

## RESEARCH MEMORANDUM

# TOTAL FORCE ENLISTMENT PROGRAMS SIMULATION

Volume II: Appendixes

Timothy W. Cooke  
George R. Corliss  
Kevin B. Garvey  
James E. Grogan

AD-A194 187

DTIC  
ELECTE  
JUN 13 1988  
S H D

A Division of

**CNA**

Hudson Institute

## CENTER FOR NAVAL ANALYSES

4401 Ford Avenue • Post Office Box 16268 • Alexandria, Virginia 22302-0268

### DISTRIBUTION STATEMENT A

Approved for public release;  
Distribution Unlimited

8 6 8 021

REPORT DOCUMENTATION PAGE				Form Approved OMB No. 0704-0188	
1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION / AVAILABILITY OF REPORT  APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED		
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE					
4. PERFORMING ORGANIZATION REPORT NUMBER(S) CRM 87-94			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION Center for Naval Analyses		6b. OFFICE SYMBOL (if applicable) CNA	7a. NAME OF MONITORING ORGANIZATION Office, Chief of Naval Operations (OP-01)		
6c. ADDRESS (City, State, and ZIP Code) 4401 Ford Avenue Alexandria, Virginia 22302-0268			7b. ADDRESS (City, State, and ZIP Code) Navy Department Washington, D.C. 20350-2000		
8a. NAME OF FUNDING / SPONSORING ORGANIZATION Office of Naval Research		8b. OFFICE SYMBOL (if applicable) ONR	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER N00014-87-C-0001		
8c. ADDRESS (City, State, and ZIP Code) 800 North Quincy Street Arlington, Virginia 22217			10. SOURCE OF FUNDING NUMBERS		
			PROGRAM ELEMENT NO. 65154N	PROJECT NO. R0148	TASK NO.
11. TITLE (Include Security Classification) Total Force Enlistment Programs Simulation: Appendixes, Volume II					
12. PERSONAL AUTHOR(S) Timothy W. Cooke, George R. Corliss, Kevin B. Garvey, James E. Grogan					
13a. TYPE OF REPORT Final		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Year, Month, Day) June 1987	
				15. PAGE COUNT 298	
16. SUPPLEMENTARY NOTATION					
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP	Calculations, Computer programs, Computerized simulation, Enlisted personnel, Manpower, SELRES (Selected Reserve), Tables (Data), Tabulation processes, User manuals		
05	09				
12	05				
19. ABSTRACT (Continue on reverse if necessary and identify by block number)  This research memorandum, CRM 87-94 (Vol. II), presents technical appendixes to supplement CRM 87-93 (Vol. I), Total Force Enlistment Programs Simulation. It is divided into four sections. Appendix A contains a guide to assist users of the simulation methodology. In addition to a detailed example of the use of the simulation, it discusses some of the potential uses not explored in Volume I. Appendix B is a description of the methods used in computing the active duty inventories and continuation behavior. It also lists some of the important active-duty data elements used in the simulation, as well as the major computer programs used to generate these data. Appendix C presents similar descriptions for the SELRES data. A flowchart of the simulation methodology and annotated versions of the simulation programs appear in appendix D.					
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL			22b. TELEPHONE (Include Area Code)		22c. OFFICE SYMBOL



# CENTER FOR NAVAL ANALYSES

A Division of Hudson Institute

4401 Ford Avenue • Post Office Box 16268 • Alexandria, Virginia 22302-0268 • (703) 824-2000

24 August 1987

## MEMORANDUM FOR DISTRIBUTION LIST

Subj: Center for Naval Analyses Research Memorandum 87-94

Encl: (1) CNA Research Memorandum 87-94, "Total Force Enlistment Programs Simulation, Vol II," Jun 1987

1. The Center for Naval Analyses (CNA) was asked to address the question of how changing the distribution of accessions among active duty enlistment programs influences future force levels in the active and selected reserve components. Enclosure (1) contains the technical appendixes to the CNA analysis.

2. The Total Force Enlistment Programs simulation is based on continuation and affiliation behavior of individuals in six enlistment programs as observed in recent years. Total accessions and their distribution among enlistment programs and ratings are inputs to the simulation. The effects of changing these inputs are simulated by comparing the future inventories associated with alternative accession profiles. Major data elements and simulation software are presented in enclosure (1).

A handwritten signature in dark ink, appearing to read "Robert J. Ravera".

ROBERT J. RAVERA  
Director  
Naval Planning, Manpower and  
Logistics Division

Distribution List:  
Reverse page

Subj: Center for Naval Analyses Research Memorandum 87-94

Distribution List

SNDL

E3D5 NAVPERSRANDCEN

FF38 USNA

Attn: Nimitz Library

FF42 NAVPGSCOL

OPNAV

OP-913

OP-01B2

OP-135

OP-135D



# **TOTAL FORCE ENLISTMENT PROGRAMS SIMULATION**

Volume II: Appendixes

Timothy W. Cooke  
George R. Corliss  
Kevin B. Garvey  
James E. Grogan

*Naval Planning, Manpower, and Logistics Division*

A Division of



Hudson Institute

---

## **CENTER FOR NAVAL ANALYSES**

4401 Ford Avenue • Post Office Box 16268 • Alexandria, Virginia 22302-0268

## ABSTRACT

This research memorandum, CRM 87-94 (Vol. II), presents technical appendixes to supplement CRM 87-93 (Vol. I), Total Force Enlistment Programs Simulation. It is divided into four sections. Appendix A contains a guide to assist users of the simulation methodology. In addition to a detailed example of the use of the simulation, it discusses some of the potential uses not explored in volume I. Appendix B is a description of the methods used in computing the active-duty inventories and continuation behavior. It also lists some of the important active-duty data elements used in the simulation, as well as the major computer programs used to generate these data. Appendix C presents similar descriptions for the SELRES data. A flowchart of the simulation methodology and annotated versions of the simulation programs appear in appendix D.



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

# TABLE OF CONTENTS

	<u>Page</u>
<b>Volume I (CRM 87-93)</b>	
<b>List of Tables .....</b>	<b>v</b>
<b>Introduction .....</b>	<b>1</b>
<b>Outline of Simulation Methodology .....</b>	<b>2</b>
<b>Active-Duty Simulation .....</b>	<b>3</b>
Computation of Transition Behavior .....	3
Computation of Transition Behavior for Accessions .....	8
Computation of Other Transition Behavior .....	9
Application of Continuation Behavior .....	9
Active-Duty Losses .....	9
Comparison of Active Inventory to Requirements .....	10
<b>SELRES Simulation .....</b>	<b>11</b>
Accessions .....	11
Computation of Continuation Behavior .....	12
Application of Continuation Behavior .....	13
Comparison of Reserve Inventory to Requirements .....	13
<b>Observed Continuation and Affiliation Behavior .....</b>	<b>13</b>
<b>Simulation Examples .....</b>	<b>17</b>
Base Simulation .....	18
Base-Simulation Results .....	21
Alternative 1: Accessions Required To Reach FY 1991 EPA ....	22
Alternative 2: Shift of 3,000 Accessions From 4YO to AM ....	23
Alternative 3: Shift of Sea College to 4YO Enlistments ....	24
Alternative 4: Shift of 3,000 Accessions From 4YO to Prior-Service .....	25
<b>Improving the Simulation .....</b>	<b>27</b>
<b>Conclusion .....</b>	<b>28</b>
<b>References .....</b>	<b>31</b>

## TABLE OF CONTENTS (Continued)

	<u>Page</u>
Volume II (CRM 87-94)	
Appendix A: User's Guide to the Total Force Enlistment Programs Simulation .....	A-1 - A-12
Appendix B: Active-Duty Data and Calculations .....	B-1 - B-100
Annex B-1: Program Listing to Extract and Tabulate Enlisted Master Record Inventory .....	B-101 - B-116
Annex B-2: APL Program Listings for Computing Aggregate Transition and Addition Rates .....	B-117 - B-125
Appendix C: SELRES Data and Calculations .....	C-1 - C-26
Annex C-1: Program Listings for Computing SELRES Affiliation and Continuation Rates .....	C-27 - C-41
Annex C-2: Program Listings for Tabulating SELRES Inventory .....	C-43 - C-60
Appendix D: Simulation Programs .....	D-1 - D-10
Annex D-1: Listing of Command Procedure to Execute Simulation Programs .....	D-11 - D-12
Annex D-2: Listing of Active-Duty Simulation Program (VAX-11 FORTRAN) .....	D-13 - D-60
Annex D-3: Listing of Reserve (SELRES) Simulation Program (VAX-11 FORTRAN) .....	D-61 - D-93

**APPENDIX A**

**USER'S GUIDE TO THE TOTAL FORCE ENLISTMENT PROGRAMS SIMULATION**

## APPENDIX A

### USER'S GUIDE TO THE TOTAL FORCE ENLISTMENT PROGRAMS SIMULATION

This appendix provides information on how to use the Total Force Enlistment Programs Simulation. For the most part, its use is self explanatory. The example and discussion below is intended to reduce the ambiguities that may arise regarding how to pose a research or policy question so that the simulation methodology gives an appropriate answer.

The Total Force Enlistment Programs simulation is a menu-oriented procedure that gives the user a sequence of choices regarding inputs to the simulation. The outputs of the simulation are files in the current directory of the user describing the inputs chosen by the user, resulting inventories, and comparisons between results of two simulations. These files must be directed to the printer by users desiring hard copy versions of the results. Alternatively, users located at a remote site employing a modem hook-up can list and capture or print these files.

The user of the simulation will need at least 1,500 blocks of available disk storage on the CNA unclassified computer system to be sure of having enough space to run both the active and reserve simulations and to compare the results of two alternative runs. However, the simulation may be run successfully with as few as 800 blocks of available storage. The simulation is best accessed through a command procedure called "SIM" that appears in appendix D. Upon typing "SIM" at the prompt, the user will be able to execute the various programs that comprise the simulation methodology. The following example illustrates the use of the simulation.

#### USING THE SIMULATION

After starting the simulation, the following screen will appear. To exit the program during the course of any simulation requires using the interrupt command 'ctrl-y'. Use of this interrupt will cause screen 1 to reappear.

- 1 - Execute Active simulation program
- 2 - Execute Reserve simulation program
- 3 - Execute program to compare two Active runs
- 4 - Execute program to compare two Reserve runs
- 9 - Finish this session

Enter your selection \_\_

SCREEN 1



On selecting option 1, the user executes the active duty simulation. When finished, the user will confront the same menu of choices again. The active duty simulation is discussed first, followed by the reserve simulation.

#### ACTIVE FORCE SIMULATION

Hit return to start run

#### SCREEN 2

The next choice that the user has is which allocation profile to use in the simulation. The default allocation of accessions to enlistment programs and ratings is derived from the FY 1985 PRIDE data, with the exception of the Prior Service program. Most applications will utilize a user-created allocation matrix (option 2) that has been customized to reflect an accession profile that the user wants to

- 1 - Use the default allocation matrix
- 2 - Selected a user created allocation matrix

Enter your selection:

#### SCREEN 3

analyze. When option 2 is chosen, a numbered list of previously created allocation matrices in the default directory appears on the screen. These files must have been created in previous runs of the simulation. An example appears below.

- 1 - AMPLUS4
- 2 - BASE
- 3 - BASE\_88A
- 4 - DEFAULT\_85P
- 5 - NEW1
- 6 - NO\_SE\_CLG

Enter your selection:

#### SCREEN 4

The percentage of accessions in each enlistment program (except Navy Sea College) and rating group is determined by this choice. The next user input is the total number of accessions in each year, less any Navy Sea College accessions. The simulation treats these as accessions rather than end-of-fiscal-year inventories, and requests inputs for the total number of accessions by year as follows:

	1 - Current FY total recruits 1986	___
	2 - Outyear 1 total recruits 1987	___
	3 - Outyear 2 total recruits 1988	___
	4 - Outyear 3 total recruits 1989	___
Total accessions	5 - Outyear 4 total recruits 1990	___
minus Sea College	6 - Outyear 5 total recruits 1991	___
	7 - Outyear 6 total recruits 1992	___
	8 - Outyear 7 total recruits 1993	___
	9 - Outyear 8 total recruits 1994	___
	10 - Outyear 9 total recruits 1995	___

Enter number of outyear to change or <Return>

#### SCREEN 5

LOS 1 Sea College program end-of-fiscal-year survivors, rather than accessions, are input similarly. For each simulation reported in volume I, 90 percent of Sea College accessions are assumed to survive to the end-of-fiscal-year.

	1 - Current FY total recruits 1986	___
	2 - Outyear 1 total recruits 1987	___
	3 - Outyear 2 total recruits 1988	___
	4 - Outyear 3 total recruits 1989	___
Inputs for	5 - Outyear 4 total recruits 1990	___
Sea College	6 - Outyear 5 total recruits 1991	___
	7 - Outyear 6 total recruits 1992	___
	8 - Outyear 7 total recruits 1993	___
	9 - Outyear 8 total recruits 1994	___
	10 - Outyear 9 total recruits 1995	___

Enter number of outyear to change or <Return>

#### SCREEN 6

At this point, the allocation percentages by enlistment program and rating are applied to the total inputs, yielding the number of accessions in each enlistment program and rating group. As a check on the feasibility of the allocation by rating group, the user can request

(on screen 7) to compare the proposed allocation of accessions by rating group to planned (FY 1988) allocations of A-school seats to USN recruits.

Would you like to view the comparison of allocations to planned school seats (Yes or No)? \_\_\_\_

#### SCREEN 7

Answering this question 'Y' yields a presentation like the one in screen 8. As presently implemented, this display is only for the user's information in connection with potential reallocations of accessions across rating groups. It can simply be ignored if the user desires. For the example presented below, the allocation of FY 1988 accessions to the AB rating group is 223 short of the planned seats, a difference of 21 percent.

No school req 7800 AN	No school Req 3200 DM	-93 -3% 2200 MS
-223 -21% 6700 AB	-23 -7% 1900 DP	-1 -1% 3300 MU
-1 0% 6600 AC	-2 0% 1010 DS	No school req 1400 NC
-167 -5% 6080 AF	-253 8% 4100 EM	-52 -2% 0300 OS
0 0% 7100 AG	-6 0% 3800 EN	0 0% 0450 OT
-30 -4% 7300 AK	-2 0% 5380 EQ	-23 -16% 2700 PC
-5 0% 6500 AO	-337 -9% 1000 ET	-3 -1% 7600 PH
-1 0% 7500 AS	-90 -17% 0350 EW	0 0% 1080 PI
-155 -5% 6280 AV	No school req 5000 FN	0 0% 4600 PM
-44 -7% 6400 AW	-77 -3% 0800 FT	-51 -8% 1800 PN
-2 0% 7400 AZ	-5 0% 0600 GM	-1 0% 7000 PR
No school req 0100 BM	-90 -14% 4400 GS	-1 0% 0200 QM
-2 0% 4000 BT	-7 0% 8000 HM	-145 -5% 1500 RM
No school req 6000 CN	1147 154% 4300 HT	0 0% 2500 RP
0 0% 1622 CTA	-2 0% 2300 IS	No school req 3600 SN
-2 0% 1666 CTI	-3 -3% 2600 JO	-3 0% 2490 SH
0 0% 1633 CTM	0 0% 3100 LI	-1 0% 2000 SK
-10 -4% 1644 CTO	No school req 1750 LN	-1 0% 0250 SM
-1 0% 1655 CTR	No school req 0150 MA	-111 -5% 0400 ST
-2 0% 1611 CTT	0 0% 4700 ML	No school req 7200 TD
-58 -4% 5080 CU	-243 -5% 3700 MM	-7 -1% 0500 TM
-2 0% 8300 DN	0 0% 0900 MN	0 0% 5800 UT
-31 -10% 2100 DK	-1 0% 3900 MR	-69 -5% 1700 YN

#### SCREEN 8

To alter these accession profiles by enlistment program or rating group for any year, the user hits <Return> after viewing the above comparison, and answers 'Y' to the following question.

Would you like to change Program or Rating mixes for any year  
(Yes or No)? \_\_\_\_

#### SCREEN 9

The year for which the accession profile is to be changed is then input in response to the following screen.

- 1 - Current FY 1986
- 2 - Outyear 1 1987
- 3 - Outyear 2 1988
- 4 - Outyear 3 1989
- 5 - Outyear 4 1990
- 6 - Outyear 5 1991
- 7 - Outyear 6 1992
- 8 - Outyear 7 1993
- 9 - Outyear 8 1994
- 10 - Outyear 9 1995

Enter your selection or (99 to end): \_\_\_\_

#### SCREEN 10

If year 3 were chosen, the following would then appear.

Selection for year 1988

- 1 - Change program mixes
- 2 - Change rating mixes
- 9 - End changing this outyear

Enter your selection: \_\_\_\_

#### SCREEN 11

If a reallocation of accessions among enlistment programs is desired, the user types '1' and views screen 12:

	<u>Orig</u>	<u>Change</u>	<u>Diff</u>
1 - 4YO Program	44,512	44,512	0
2 - Active Mariner Program	13,224	13,224	0
3 - 5 & 6YO Program	14,988	14,988	0
4 - Prior Service Program	6,251	6,251	0
5 - TAR Program	1,271	1,271	0
6 - Sea College Program	2,700	2,700	0
7 - Any new program	0	0	0
Totals	82,946	82,946	0

Enter Program to change or <Return> \_\_\_\_

#### SCREEN 12

Selecting a program by number, the user may change the number of accessions as desired. The program keeps track of the differences so that changes may be analyzed more easily. The new inputs are used to recompute the allocation percentages for that year. The percentage distribution of accessions across rating groups within each enlistment program is unchanged in this process.

If a reallocation of accessions among ratings is desired, the user types '2' and views:

1. 7800 AN	20. 1611 CTT	39. 2600 JO	58. 0200 QM
2. 6700 AB	21. 5080 CU	40. 3100 LI	59. 1500 RM
3. 6600 AC	22. 8300 DN	41. 1750 LN	60. 2500 RP
4. 6080 AF	23. 2100 DK	42. 0150 MA	61. 3600 SN
5. 7100 AG	24. 3200 DM	43. 4700 ML	62. 2490 SH
6. 7300 AK	25. 1900 DP	44. 3700 MM	63. 2000 SK
7. 6500 AO	26. 1010 DS	45. 0900 MN	64. 0250 SM
8. 7500 AS	27. 4100 EM	46. 3900 MR	65. 0400 ST
9. 6180 AV	28. 3800 EN	47. 2200 MS	66. 7200 TD
10. 6400 AW	29. 5380 EQ	48. 3300 MU	67. 0500 TM
11. 7400 AZ	30. 1000 ET	49. 1400 NC	68. 5800 UT
12. 0100 BM	31. 0350 EW	50. 0300 OS	69. 1700 YN
13. 4000 BT	32. 5000 FN	51. 0450 OT	
14. 5000 CN	33. 0800 FT	52. 2700 PC	
15. 1622 CTA	34. 0600 GM	53. 7600 PH	
16. 1666 CTI	35. 4400 GS	54. 1080 PI	
17. 1633 CTM	36. 8000 HM	55. 4600 PM	
18. 1644 CTO	37. 4300 HT	56. 1800 PN	
19. 1655 CTR	38. 2300 IS	57. 7000 PR	

Input Rating (99 to end) \_\_\_\_

#### SCREEN 13

The user may select up to 9 ratings at a time for changes. In this example, 5 rating groups were chosen yielding the following display. To change the number of accessions in the AB rating group, 4Y0 program, the user would select row 1, column 1 and enter the new number. The total is automatically adjusted. If the user changes only the total column

	<u>4Y0</u>	<u>ACT MA</u>	<u>5&amp;6Y0</u>	<u>PR SER</u>	<u>TARS</u>	<u>SEA CO</u>	<u>OTHER</u>	<u>TOTAL</u>	<u>MAX</u>
6700 AB	696	0	0	126	0	0	0	822	1,045
6400 AW	383	93	0	69	36	0	0	581	625
1611 CTT	265	0	0	15	0	0	0	280	282
1000 ET	0	211	2,772	123	68	0	0	3,174	3,511
3100 LI	33	0	0	8	0	0	0	41	41

Enter ROW and COLUMN to change or <Return>      ROW \_\_\_\_ COLUMN \_\_\_\_

#### SCREEN 14

for a rating, the new total will be distributed across enlistment programs in the same proportion as the original total. The MAX column



contains the number of A-school seats allocated for FY 1988, and is provided for information only. These procedures may be executed as many times as desired, until the user is satisfied with the allocation of accessions in the chosen year. On exiting from the changes (for all years) the user is given the opportunity to save the changes.

Would you like to save this allocation (Yes or No)? Y

Enter a name to save this allocation under \_\_\_\_\_.

#### SCREEN 15

The version of the simulation using aggregate data provides three alternative years of continuation behavior. They are chosen from the following menu.

Choose the transition rates you desire for this run

- 1 - Transitions from 80 to 81
- 2 - Transitions from 84 to 85
- 3 - Transitions from 85 to 86

Enter your selection:

#### SCREEN 16

The version of the simulation using individual data presently provides only FY 1986 continuation behavior.

The simulation now executes as shown on screen 17, producing a file of simulated inventories that will be saved with a name chosen by the user. It is stored in binary format for use by the COMPARE procedure.

Please wait while simulation takes place

Beginning to transition the history  
Finished creating the outyear inventories

Beginning to add paygrade to outyear inventories  
Finished adding paygrade to outyear inventories

Beginning to write results to disk  
Finished writing results to disk

Enter a filename for storing the results (for use later in the COMPARE run) \_\_\_\_\_.

#### SCREEN 17

A second file called ACTIVE.DAT is written to disk and may be printed to see the allocation of new accessions and resulting simulated inventories for each year. A file of active duty losses by rating group and enlistment program is also produced for use in the reserve simulation. This finishes the active duty simulation.

Finish of Active Force Simulation

Hit return to finish

#### SCREEN 18

The user now chooses among the options on SCREEN 1. Choosing to run the reserve simulation (option 2) brings up the following screen.

Reserve Force Simulation

Hit return to start run

#### SCREEN 19

The total number of SAM accessions are entered on screen 20. Like active duty accessions, these should not be adjusted for within-year attrition. As described in volume I, the simulation takes 90 percent of these numbers as LOS 1 survivors.

	1 - Current FY total recruits:	_____
	2 - Outyear 1 total recruits:	_____
	3 - Outyear 2 total recruits:	_____
	4 - Outyear 3 total recruits:	_____
Input for SAMs	5 - Outyear 4 total recruits:	_____
	6 - Outyear 5 total recruits:	_____
	7 - Outyear 6 total recruits:	_____
	8 - Outyear 7 total recruits:	_____
	9 - Outyear 8 total recruits:	_____
	10 - Outyear 9 total recruits:	_____

Enter number of outyear to change of <Return>

#### SCREEN 20

The residual category of OTHER SELRES accessions is input on a screen similar to screen 20. Like Sea College accessions, these should include end-of-fiscal-year LOS 1 survivors.

The user may now change the rating mix of SAM accessions by answering 'Y' to screen 21. This procedure is the same as changing rating distributions for the active simulation, except that only SAM accessions may be reallocated.

Would you like to change SAM mixes (Yes or No)?

#### SCREEN 21

The reserve simulation then executes, writing the results to two disk files. The user is asked to name a binary version of the output file for use in later comparisons. A second version called RESERVE.DAT is produced and may be printed or edited to examine the simulated inventories.

Enter a filename for storing the results  
(for use later in the COMPARE run) \_\_\_\_\_

#### SCREEN 22

The reserve simulation is now finished.

Finish of reserve force simulation

Hit return to finish

### SCREEN 23

A program to compare the results of two simulations may then be executed by choosing options 3 or 4 on screen 1.

This will cause a list of existing active duty or reserve simulation output files to be displayed as on screen 24. Selection of two files by number yields a detailed comparison by paygrade and rating group for each outyear. This is written to disk as COMPARE\_ACT.DAT or COMPARE\_RES.DAT.

- 1 - AMPLUS4
- 2 - BASE1
- 3 - BASE\_88A
- 4 - EXAMPLE
- 5 - NO\_SEA\_C G

Enter selection for base file: \_\_ Compare file: \_\_

### SCREEN 24

The primary use of the simulation is to examine the effects of changing the mix of active duty accessions among enlistment programs. The easiest way to do this is to use the options described on screens 9-12. However, this procedure will generally shift accessions among rating groups. This happens because ratings do not have accessions in each of the enlistment programs and the allocation of accessions to rating groups within an enlistment program is fixed by the allocation matrix in use. For example, in alternative 4 of volume I, accessions are redistributed from the 4Y0 program to the Prior-Service program. The 5Y0 ratings have no 4Y0 enlistments but do have Prior-Service enlistments. Such ratings will grow as the Prior-Service program grows relative to the 4Y0 program. The user may circumvent this redistribution among ratings by reallocating accessions among enlistment programs for each of the ratings, making sure to keep rating sizes fixed in the process.

For the Sea College program, the number of accessions must be changed on screen 6 to have the effect consistently included in the simulation. In fact, the simulation will not allow the user to change

the number of Sea College accessions on screen 12. This is because the Sea College program allocations to the General Detail (GENDET) categories, based on the distribution of GENDETs among the Seaman, Airman, and Fireman categories, is done outside the allocation process used for other enlistment programs.

A second use of the simulation is long-term planning for the number of accessions required to meet a given inventory objective by a certain time. This can be accomplished by varying the total accessions until the appropriate inventory level is reached. In the process, the effect of changes in the number of active duty accessions on SELRES inventories can be estimated.

Using continuation rates associated with aggregate data, the simulated effects of redistributing accessions among rating groups are likely to be misleading because of the fact that only net continuation behavior is observed. That is, lateral transfers between ratings are not tracked in this approach. The effects will be particularly misleading when only the first three years of continuation are observed, because GENDETs make lateral transfers to rating groups at relatively high ratings during the first three years of service. For this reason, the net continuation rates for the ratings are not representative of individuals beginning service with initial training in those ratings. The continuation rates based on individual data do not pose this difficulty, and be used to more reliably estimate changes in a rating group associated with changing the number or mix of enlistments into that rating. Such use of the simulation should be pursued cautiously because of the relative validity of rating sizes and the impact of unpredictable short-term corrective actions applied to particular ratings.

**APPENDIX B**

**ACTIVE-DUTY DATA AND CALCULATIONS**



## APPENDIX B

### ACTIVE-DUTY DATA AND CALCULATIONS

The Total Force Enlistment Program simulation requires information on historical inventories and continuation behavior of enlisted personnel by program of enlistment, rating, and length of service. This information was obtained by extracting end-of-fiscal-year inventories from CNA Enlisted Master Record (EMR) files and storing the information in two types of data tables. For aggregate continuation behavior, and inventory calculations, counts of individuals by program of entry, rating, paygrade, and length of service (LOS) are produced. Tables for each fiscal year 1979 to 1986 were loaded into APL files as matrices. For individual continuation behavior, individual observations for the end of fiscal years 1985 and 1986 are summed into arrays by enlistment program, rating, transition type, and LOS. The transition types include within rating continuation, lateral transfer into and out of the begin-year rating, and loss from active duty.

The advantage of storing the yearly data as APL matrices is that each years' inventory, as a single data object, can be manipulated--summed or extracted across the four dimensions--by single APL operations. Also, year-to-year comparisons, like transition rates, can be performed on whole matrices at a time rather than iterative operations that would perform the operations cell-by-cell.

The program shown in annex B-1 extracts inventory data from the EMRs. Standard methods are used in the program to determine rating, paygrade, and LOS. Individuals are initially divided into 129 ratings, although later analysis required that they be merged into 69 rating groups. These 69 rating groups correspond to the separate paths available to enlisted personnel advancing into chief (E7 to E9) ratings. The paygrades are the nine pay levels E1 through E9. LOS is determined by using the Active Duty Service Date (ADSD). Individuals are placed in one of 31 LOS categories, 1 through 30, and 31 and greater years of service, by measuring time between ADSD and the date of the end-of-FY EMR snapshot.

The entry program is determined primarily by the PROGRAM-ENLISTED-FOR field in the SPECIAL-PROGRAM-CODES (SPC) section of the EMR, although other criteria were used when SPC code was missing. Five enlistment entry programs were defined for those with length of service of zero to nine years: four-year obligor (4YO), Active Mariner (AM), five/six-year obligor (5/6 YO), Prior-Service (PS), and TAR Enlistment Program (TAR) (enlisted personnel of LOS ten years and greater were not divided into entry programs). Enlisted personnel who could not be classified by entry program were excluded from the analysis. This was more likely for the higher LOS cells of earlier inventory years, corresponding to accessions during a time when enlistment programs were

not as frequently recorded. By 1985, only about 50 individuals are excluded. Figure B-1 presents a flow chart of the procedure used to determine original enlistment program for each individual. The special program and type acquisition codes used for classification are in table B-1.

TABLE B-1  
ENLISTMENT PROGRAM CLASSIFICATION CODES

<u>Enlistment program</u>	<u>SPC</u>	<u>Type acquisition</u>
4YO	H 4 F K E	18, 48
Active Mariner	M Z W	
6YO	G 5 A B	
Prior-Service	S 8 7 N Q L	19
TAR	Y	

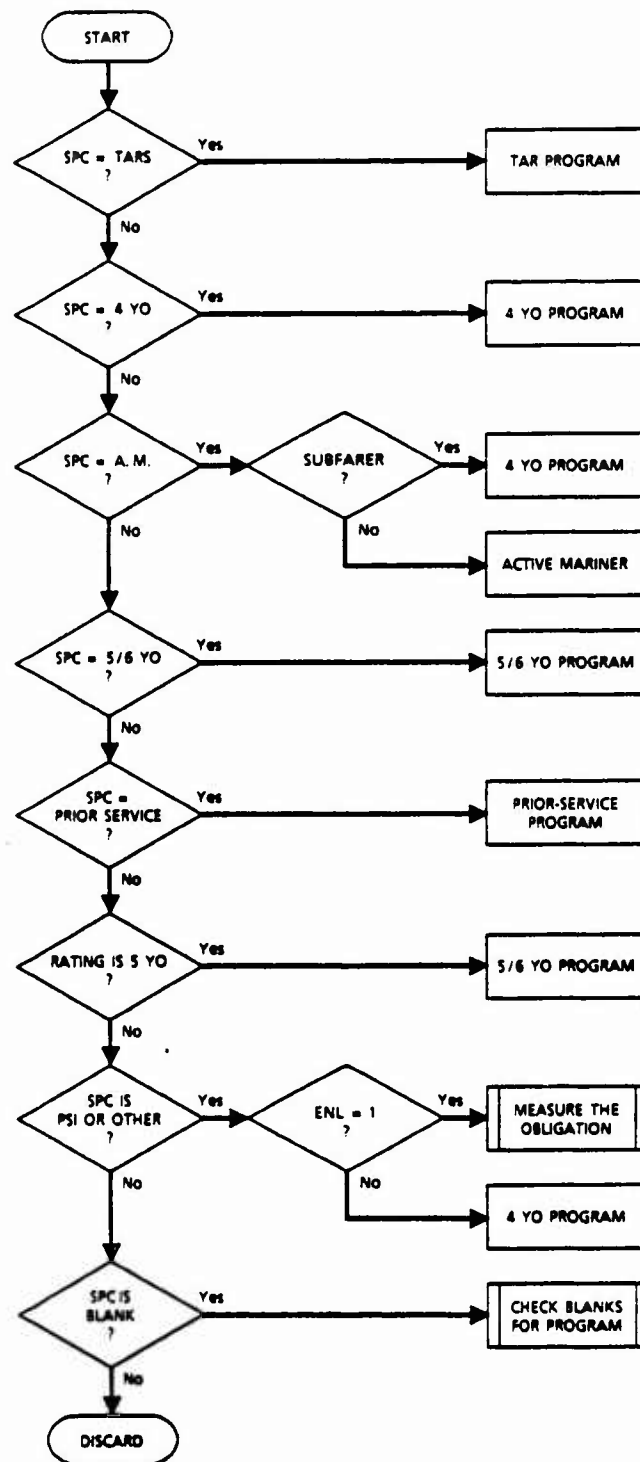
The APL inventory matrices are stored in the APL file, [GARVEYK.APL]SEPTEMBER.AIX. Other historical data, such as PRIDE data, and new accession prior service, are also necessary in producing data for the model and are stored as matrices of identical shape as the main inventory data. They also reside in APL files. APL files are a collection of "components" that are referenced by number. All APL files used in the analysis have the same organization: an explanatory index as the first component, with the data stored by year in components two through nine, starting with 1986 data in component two, 1985 in three, and so on, in reverse chronological order.

#### Continuation Behavior From Aggregate Data

Given the EMR inventories and other historical data tabulated by enlistment program, rating and LOS, aggregate continuation behavior can be computed. Annex B-2 lists the annotated APL programs used for this purpose. The function XTRANS\_TO computes transition and addition rates for each combination of enlistment program, rating group, and LOS for any two adjacent years for which inventories are available. The earliest such year is fiscal 1980 and the latest year is fiscal 1985.

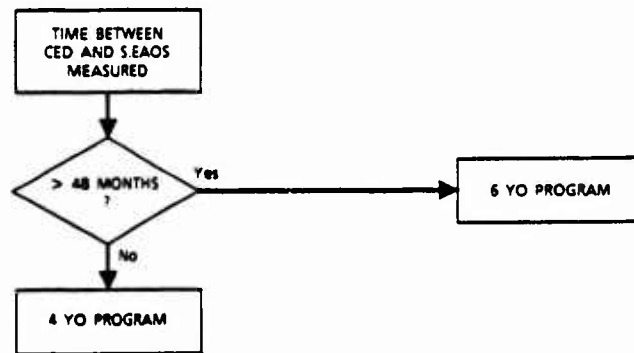
---

1. Computation of continuation rates by enlistment program reveals that some simple continuation rates exceed 1.0 in the early LOS cells. This must be due to changes in the program-enlisted-for field of the EMR when individuals switch from regular active duty to TAR. A detailed investigation of these changes is left for future research.



**FIG. B-1: FLOWCHART OF PROCEDURE TO DETERMINE ENLISTMENT PROGRAM**

Measure the obligation



Check blanks for program

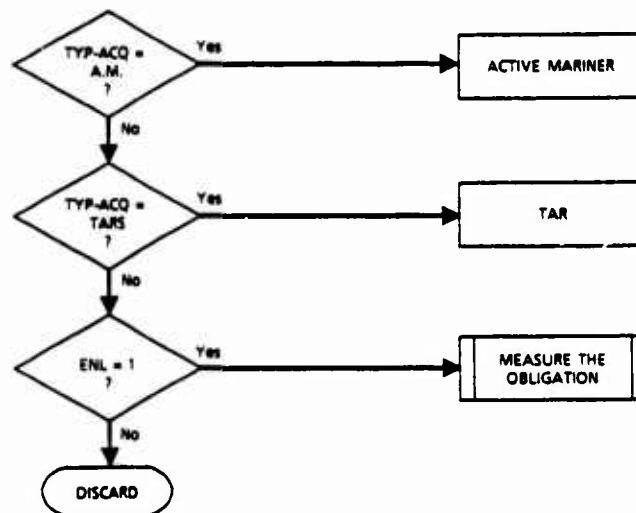


FIG. B-1: (Continued)

The inventory tabulations are prepared for the computation of transition and addition rates by the functions BUILD LOS1 and MAKE ALLOCATION. PRIDE data are used to distribute the LOS1 inventory by rating and enlistment program. In this process, all prior service accessions are placed in LOS1, to be spread back out over the appropriate LOS cells in the simulation. For the LOS1 to LOS3 transitions, the results of BUILD\_LOS1 are used in the calculations of XTRANS\_TO.

After setting up the data, the first step in the calculation of aggregate transition rates is to subtract new Prior Service accessions from the inventory of the end-year to produce the numerator of the transition rate. This is necessary because including new Prior-Service accessions by LOS would exaggerate transition rates for that cell by counting accession behavior as continuation. Following this, the function TRANS\_DIVIDE\_BY divides the end-year inventory by the begin-year inventory and produces both the transition and addition rates. Where transition rates not exceeding one are computed, the addition rate is zero. Otherwise, the addition rate is calculated as the excess in the end-year (over the begin-year) divided by the size of the rating in the begin-year. In the aggregate transition version of the simulation, the addition rate is multiplied by the size of the rating to account for lateral transfers. For LOS3 additions, the denominator of the addition rate is the size of the total LOS1 inventory one year prior to the begin year.

As it turned out, the large number of lateral rating transfers made the results difficult to interpret without collapsing the rating dimension. For this reason, the analysis resorted to calculation of continuation behavior (keeping the rating dimension) from individual observations.

#### Continuation Behavior from Individual Behavior

The calculation of continuation rates using individual observations is based on EMR files from September 1985 and 1986. Each individual with LOS 3 or greater is identified on the September 1985 EMR according to enlistment program, rating and LOS. Those categories are established on the same criteria as the inventory tabulation. The status of that individual is then observed on the September 1986 EMR. The individual's transition is classified as (1) a loss to active duty endstrength, (2) continuation in the same rating, or (3) continuation with a change of rating. The rates of occurrence of each transition type are tabulated for each cell of the inventory matrix. LOS 1-3 transitions are derived from comparing FY 1984 PRIDE accessions to FY 1986 inventories. Table B-2 contains the end-of-fiscal-year 1985 inventory of enlisted personnel categorized by enlistment program, rating, and LOS. Rates for the three types of continuation behavior are listed in tables B-3 through B-5. Within rating continuation rates are in table B-3. The lateral transfer rates by enlistment program, rating,

TABLE B-2

FY 1985 INVENTORY BY  
ENLISTMENT PROGRAM, RATING AND LOS

## 4YO PROGRAM

LOS	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	4814	668	0	2500	137	378	735	227	1671	418	364	0	499	0
2	3579	447	58	2200	136	349	642	192	1497	274	313	76	139	9
3	1075	533	70	2246	111	331	647	240	1533	270	342	472	494	5
4	564	503	53	1976	110	338	534	218	1712	266	328	747	550	0
5	130	345	112	1254	75	259	362	160	862	167	234	626	207	0
6	21	306	68	1160	66	244	224	132	836	113	186	370	222	0
7	14	270	93	1076	53	216	194	96	692	76	155	254	381	0
8	6	230	117	929	41	144	171	64	543	90	116	203	314	0
9	3	220	100	733	44	157	167	54	309	83	80	232	289	0

LOS	1622 CTA	1666 CTT	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	132	0	0	208	722	0	0	0	226	0	0	0	1133	1327
2	83	95	9	121	194	136	51	47	176	1	39	5	912	862
3	73	88	31	225	195	191	32	83	230	21	135	8	1393	894
4	91	70	23	235	123	142	33	80	267	21	155	9	1097	850
5	69	47	22	105	76	108	20	83	244	23	190	16	649	510
6	46	39	13	86	56	78	19	38	210	21	97	26	527	395
7	47	29	16	56	71	113	14	42	119	15	80	24	389	342
8	49	35	9	64	93	62	16	54	57	22	90	55	382	189
9	50	30	18	56	57	48	43	58	61	14	75	74	254	125



TABLE B-2 (Continued)

## 4YO PROGRAM

LOS	5300 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0	0	181	4525	0	1203	0	3081	1479	208	0	0	0	0
2	52	54	219	3507	71	641	18	3321	1326	140	12	5	0	2
3	28	137	255	1138	127	568	24	2733	1292	124	19	25	2	2
4	41	104	232	539	195	476	213	2124	1095	95	25	27	7	1
5	39	150	101	73	175	290	122	1536	483	53	30	37	23	11
6	23	96	51	19	151	232	70	1447	395	46	23	33	26	18
7	17	78	18	9	62	206	54	976	369	39	15	17	15	24
8	17	42	9	6	50	198	38	692	243	50	16	14	39	28
9	23	110	7	3	66	230	54	461	232	44	20	19	19	27

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	26	718	54	273	1411	0	0	1410	243	104	0	116	24	476
2	16	273	51	254	1253	29	0	1118	144	125	10	91	13	411
3	23	1097	38	372	1179	56	0	1300	117	119	39	124	20	438
4	19	901	35	341	1013	80	0	812	167	129	19	117	18	539
5	11	478	21	146	720	49	0	550	131	71	25	73	6	507
6	6	480	21	109	618	17	0	304	122	54	30	41	3	419
7	6	508	14	83	476	13	5	225	56	38	23	20	3	302
8	3	427	23	55	429	24	7	224	49	29	23	22	1	276
9	1	440	17	24	440	39	20	250	32	15	32	15	3	250

LOS	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	184	596	2619	134	11530	326	770	454	413	0	339	0	898	49954
2	162	472	1749	107	7082	487	525	301	471	0	217	33	754	38158
3	183	410	1944	100	3009	461	809	265	316	48	309	36	1051	32275
4	160	472	1728	97	1432	408	772	175	206	128	291	31	1080	27439
5	109	230	986	39	267	203	488	163	170	35	189	31	825	16601
6	70	121	764	43	66	144	448	84	134	17	128	21	643	13136
7	87	136	612	40	22	141	276	47	73	19	141	12	471	10675
8	55	123	419	24	21	136	252	68	67	27	141	13	366	8891
9	57	78	345	17	14	97	266	95	94	30	154	19	270	7863

TABLE B-2 (Continued)

## ACTIVE MARINER PROGRAM

LOS	7800	AN	6700	AB	6600	AC	6080	AF	7100	AG	7300	AK	6500	AO	7500	AS	6180	AV	6400	AW	7400	AZ	0100	BM	4000	BT	6000	CN
1	2121		0		94		618		43		31		224		33		363		111		11		0		141		0	
2	1052		26		58		593		45		55		193		47		361		71		35		29		133		2	
3	804		188		75		760		56		67		182		96		562		79		84		318		198		2	
4	159		121		36		440		18		66		168		88		350		61		62		340		113		0	
5	24		83		31		309		14		73		128		40		268		33		46		182		82		0	
6	14		68		18		372		26		63		74		37		268		34		31		144		80		0	
7	2		50		17		410		12		62		71		37		235		41		25		135		123		0	
8	3		43		14		322		6		50		62		33		178		65		28		84		53		0	
9	1		29		12		234		6		39		58		19		156		34		25		59		18		0	

LOS	1622	CTA	1666	CTI	1633	CTM	1644	CTO	1655	CTR	1611	CTT	5080	CU	8300	DN	2100	DK	3200	DM	1900	DP	1010	DS	4100	EM	3800	EN
1	26		0		10		40		164		0		175		232		14		0		23		15		75		5	
2	21		4		6		39		18		20		58		145		15		0		28		7		91		54	
3	15		6		14		22		18		13		79		74		49		7		41		14		251		347	
4	7		1		7		5		29		12		37		35		53		4		34		11		218		216	
5	5		3		3		5		11		11		27		18		41		5		31		17		142		131	
6	6		1		5		7		1		17		29		7		21		2		25		3		144		88	
7	6		4		3		12		4		14		22		8		22		4		19		17		109		74	
8	9		3		4		3		6		11		22		12		11		3		16		11		74		52	
9	4		4		3		6		5		15		17		13		23		4		18		24		52		38	

TABLE B-2 (Continued)

## ACTIVE MARINER PROGRAM

LOS	5300 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	179	198	56	1933	191	397	0	771	427	27	11	0	0	0
2	96	93	42	998	122	223	2	657	393	21	7	2	0	0
3	154	83	67	737	131	192	9	317	450	46	14	4	0	0
4	72	36	29	140	109	163	9	233	204	39	9	9	5	1
5	19	47	18	18	57	105	5	172	150	22	6	4	2	4
6	27	48	7	9	40	73	9	194	135	14	1	4	5	6
7	22	39	3	6	23	76	6	171	124	22	12	8	6	2
8	25	30	3	2	14	33	6	78	73	10	7	1	5	6
9	17	31	4	0	13	19	13	71	26	13	10	4	1	7

B  
10

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	11	38	40	79	553	0	0	438	17	0	5	58	0	14
2	4	103	17	67	195	0	0	347	11	4	5	38	1	37
3	6	223	16	98	200	2	0	261	19	40	22	31	3	62
4	6	134	13	65	184	2	1	131	13	30	8	16	1	89
5	2	97	12	47	140	1	2	106	19	12	6	8	0	96
6	2	122	6	31	144	1	2	89	12	9	4	5	2	73
7	3	183	5	32	152	0	1	65	9	9	17	3	1	103
8	2	91	9	15	108	0	3	39	6	6	10	5	0	116
9	1	78	3	10	76	0	5	16	1	3	7	1	2	114

LOS	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	44	106	226	0	5259	18	417	117	126	0	151	84	104	16664
2	38	113	191	7	2032	20	337	85	92	0	85	58	191	9940
3	36	132	264	15	1365	33	196	63	109	3	38	73	141	10077
4	43	100	160	7	313	32	175	46	73	4	35	39	130	5599
5	17	59	122	6	61	30	134	25	61	3	28	20	132	3638
6	17	33	99	6	31	35	114	18	35	3	45	17	154	3266
7	24	35	76	7	23	53	85	14	31	2	46	9	165	3211
8	15	22	72	5	8	21	74	23	12	0	28	11	158	2327
9	18	28	62	2	3	18	71	9	11	3	12	7	147	1853

TABLE B-2 (Continued)

## 546YO PROGRAM

LOS	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0	0	470	0	0	0	0	0	1455	0	0	0	565	0
2	265	7	224	20	4	11	8	7	1188	4	9	16	490	5
3	96	16	170	54	4	16	18	13	1094	11	14	105	518	5
4	59	26	191	73	5	18	30	13	957	18	28	176	529	1
5	18	31	191	72	11	15	19	12	670	21	26	85	414	0
6	11	40	126	82	5	33	12	9	645	10	14	57	323	0
7	1	14	86	104	5	23	24	15	430	7	10	28	114	0
8	1	11	9	29	3	3	4	3	369	7	3	28	48	0
9	0	3	9	29	1	1	6	5	322	4	4	13	10	0

LOS	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0	0	315	0	0	0	1180	527	0	0	181	259	437	0
2	2	47	171	38	49	43	427	234	4	1	185	231	1299	23
3	8	44	206	25	61	54	391	201	16	1	312	351	1443	34
4	13	27	186	18	56	89	335	206	17	3	371	338	1452	53
5	7	48	205	18	47	82	517	242	20	1	412	274	1254	43
6	2	34	224	11	16	32	232	127	10	0	264	301	1108	30
7	5	8	118	15	3	12	150	155	8	1	143	121	512	31
8	3	3	101	3	4	4	137	155	2	9	95	115	477	11
9	1	5	91	4	0	4	64	69	4	1	70	91	350	8

TABLE B-2 (Continued)

## 546YO PROGRAM

LOS	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	296	2357	222	0	1909	0	272	276	125	0	71	0	0	0
2	266	2183	195	352	1557	193	275	380	133	1	57	0	0	0
3	289	2479	221	152	1275	189	406	410	132	7	60	1	0	1
4	245	2357	179	71	978	205	189	326	131	11	39	2	2	0
5	405	2160	160	14	829	99	134	306	94	7	37	2	2	2
6	197	1833	180	3	799	49	91	285	81	6	28	3	4	10
7	129	1046	106	1	434	24	52	173	36	8	23	4	3	4
8	99	924	76	2	541	15	47	116	34	9	23	1	1	1
9	59	720	63	0	303	13	70	72	40	4	11	1	2	1

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0	1256	0	0	0	0	0	0	62	0	107	0	0	0
2	0	2391	3	24	183	0	0	44	32	7	83	26	0	33
3	2	2246	1	12	339	1	0	146	23	11	101	19	0	46
4	4	2279	2	28	412	2	0	417	31	13	120	7	0	56
5	1	1857	1	6	225	5	0	482	21	6	204	5	0	51
6	1	1608	2	9	265	2	0	155	17	9	93	5	0	34
7	0	786	0	6	33	0	0	41	10	4	82	3	0	22
8	0	853	0	6	21	1	1	24	7	1	55	2	1	26
9	0	570	0	4	16	2	1	24	5	0	41	0	0	9

LOS	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0	0	282	0	0	0	0	0	1345	0	0	248	0	14217
2	3	13	311	4	1932	3	100	86	826	0	8	286	31	17033
3	11	49	315	12	705	11	103	52	691	0	55	247	36	16138
4	9	41	235	14	295	16	53	64	533	2	145	264	78	15143
5	17	30	212	9	65	19	42	45	479	3	108	323	69	13291
6	8	20	165	5	20	11	37	33	527	2	43	146	64	10608
7	4	16	111	5	14	11	23	4	292	1	25	99	42	5820
8	2	11	86	3	5	8	19	6	259	4	15	77	19	5038
9	1	6	86	1	2	4	12	2	184	2	5	38	14	3557

TABLE B-2 (Continued)

## PRIOR SERVICE PROGRAM

LOS	7800	AN	6700	AB	6600	AC	6080	AF	7100	AG	7300	AK	6500	AO	7500	AS	6180	AV	6400	AW	7400	AZ	0100	BM	4000	BT	6000	CN
1		28	5		1		3			0		1		1		0		3		2		2		2		0		0
2		9	2		2		4			0		3		9		2		16		3		0		4		5		0
3		10	4		1		14			0		4		9		1		17		8		1		3		4		0
4		41	25		9		88			4		12		46		2		59		17		9		40		29		0
5		57	58		14		162			10		42		95		22		108		45		42		129		91		0
6		24	74		22		132			8		30		139		24		177		35		43		148		109		0
7		16	150		41		498			23		73		156		37		297		32		50		228		186		1
8		6	188		57		599			40		74		173		27		381		69		70		310		205		0
9		5	117		75		384			15		70		135		34		333		68		54		262		192		0

LOS	1622	CTA	1666	CTI	1633	CTM	1644	CTO	1655	CTR	1611	CTT	5080	CU	8300	DN	2100	DK	3200	DM	1900	DP	1010	DS	4100	EM	3800	EN
1		0		0		0		1		2		0		7		5		1		0		0		0		0		2
2		0	0	0		0		1		4		0		40		0		0		0		3		0		3		10
3		0	0	1		1		2		0		0		9		0		3		0		2		0		7		9
4		0	0	1		0		4		3		3		13		6		8		0		9		3		53		40
5		1	4			4		6		14		7		21		13		13		6		9		3		148		144
6		6	11			5		13		24		17		36		23		33		7		33		5		173		150
7		10	15			14		28		27		26		63		24		42		5		65		12		251		188
8		9	23			17		55		49		43		54		36		41		3		57		23		292		222
9		24	26			23		35		34		31		38		15		32		7		48		28		207		160

TABLE B-2 (Continued)

## PRIOR SERVICE PROGRAM

LOS	5380	EQ	1000	ET	0350	EW	5000	FN	0800	FT	0600	GM	4400	GS	8000	HM	4300	HT	2300	IS	2500	JO	3100	LI	1750	LN	0150	MA
1		9		1		0		42		0		2		0		26		5		1								0
2		21		4		2		12		5		5		2		34		11		1								0
3		9		5		1		6		4		8		2		18		8		1								0
4		12		20		6		23		21		47		5		81		72		2								1
5		14		33		7		45		39		127		9		147		168		12								3
6		26		63		12		8		56		172		18		218		248		10								3
7		32		93		15		8		86		165		17		227		277		23								4
8		36		133		25		2		116		212		21		352		326		11								14
9		29		151		25		2		79		158		24		286		203		28								24
																												28

LOS	4700	ML	3700	MM	0900	MN	3900	MR	2200	MS	3300	MU	1400	NC	0300	OS	0450	OT	2700	PC	7600	PH	1080	PI	4600	PM	1800	PN
1		0		2		2		1		6		0		1		7		0		0								1
2		0		11		4		0		3		1		1		10		0		1								0
3		0		13		4		1		11		0		0		6		0		1								0
4		0		53		2		13		53		4		0		43		5		0								3
5		5		129		7		39		58		8		0		106		12		10								16
6		6		173		4		59		95		5		2		174		21		24								20
7		5		287		11		68		212		7		1		178		26		37								24
8		5		325		11		53		244		13		5		213		32		28								63
9		3		333		13		42		257		5		12		240		25		21								86
																												79

LOS	7000	PR	0200	QM	1500	RM	2500	RP	3600	SN	2490	SH	2000	SK	0250	SM	0400	ST	7200	TD	0500	TM	5800	UT	1700	YN	TOTAL	
1		0		7		17		0		320		0		8		4		3		0		0		7			2	540
2		2		6		9		1		58		2		12		4		11		0		3		18			2	377
3		1		0		21		2		37		3		6		1		6		1		1		7			4	303
4		9		27		77		1		98		15		29		15		31		2		11		5			17	1357
5		24		78		204		20		185		38		62		60		49		5		20		9			53	3095
6		28		71		256		12		55		64		116		60		66		0		28		22			48	3807
7		38		90		268		24		49		128		139		59		68		6		49		36			107	5544
8		29		105		336		16		27		158		136		85		76		13		82		26			157	6709
9		40		68		277		17		16		112		123		53		84		13		66		20			168	5639

TABLE B-2 (Continued)

## TAR PROGRAM

LOS	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0	0	19	276	0	76	60	43	297	39	45	0	57	0
2	45	1	8	103	0	57	26	17	190	18	41	0	14	0
3	17	0	10	140	0	19	9	13	172	17	12	2	18	0
4	8	1	9	139	0	16	25	8	157	11	15	6	27	0
5	0	1	2	110	0	26	28	3	94	17	11	17	19	0
6	3	1	1	94	0	20	15	7	82	10	14	21	16	0
7	0	0	1	84	0	8	12	11	42	5	2	17	16	0
8	0	0	0	51	0	10	17	5	44	6	3	11	11	0
9	0	0	1	35	0	4	4	1	22	1	2	6	7	0

LOS	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0	0	0	0	0	0	0	0	0	0	0	0	51	9
2	0	0	0	1	0	0	0	0	0	0	0	0	21	11
3	0	1	0	0	0	0	0	0	0	0	0	0	7	5
4	0	0	0	0	0	0	0	0	0	0	0	0	18	7
5	0	0	0	0	0	0	0	0	1	0	0	0	9	14
6	0	0	0	0	0	0	0	0	0	0	0	0	15	17
7	0	0	0	0	0	0	0	0	0	0	0	0	10	8
8	0	0	0	0	0	0	0	0	0	0	1	0	6	5
9	0	0	0	0	0	0	0	0	0	0	0	0	1	0



TABLE B-2 (Continued)

## TAR PROGRAM

LOS	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0	57	0	0	0	0	0	50	44	0	7	0	0	0
2	0	34	0	25	0	0	0	59	43	0	3	0	0	0
3	0	6	0	2	0	0	0	8	12	0	1	0	0	0
4	0	9	0	1	1	1	0	3	24	0	2	0	0	0
5	0	6	0	0	1	0	0	26	23	0	1	0	1	0
6	0	8	0	0	0	0	0	12	24	0	1	0	0	0
7	0	6	0	0	0	0	0	9	17	0	5	0	0	1
8	0	5	0	0	0	0	0	14	8	0	1	0	0	0
9	0	5	0	0	0	0	0	6	4	0	0	0	0	0

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0	55	0	7	4	0	0	0	0	0	3	0	0	42
2	0	14	0	8	3	0	0	2	1	0	0	0	0	36
3	0	10	0	1	0	0	0	2	1	0	1	0	0	20
4	0	24	0	3	1	0	0	0	0	0	0	0	0	16
5	0	29	0	3	0	0	0	0	0	0	0	0	0	31
6	0	20	0	5	1	0	0	1	0	0	0	0	0	56
7	0	21	0	4	1	0	0	0	0	0	1	0	0	23
8	0	8	0	4	0	0	0	0	0	0	1	0	0	26
9	0	9	0	3	1	0	0	1	0	0	0	0	0	18

LOS	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	20	0	4	0	0	0	75	0	0	0	0	0	127	1467
2	19	0	2	0	38	4	41	0	0	0	1	0	85	971
3	12	0	1	0	5	0	14	0	0	0	1	0	30	569
4	4	0	2	0	0	1	32	0	0	0	0	0	27	598
5	11	0	1	0	0	0	34	1	0	1	0	0	68	589
6	2	0	0	0	0	1	42	0	0	0	0	0	52	541
7	4	0	3	1	0	0	31	0	0	0	0	0	32	375
8	4	0	1	0	0	0	11	0	0	0	0	0	28	281
9	0	0	1	0	0	0	11	0	0	0	0	0	13	156

TABLE B-2 (Continued)

LOS &gt; 9

LOS	7800	AN	6700	AB	6600	AC	6080	AF	7100	AG	7300	AK	6500	AO	7500	AS	6180	AV	6400	AW	7400	AZ	0100	BM	4000	BT	6000	CN
10		2	275	152	979	71	218	236	86	927	166	117	386	434	0													
11		2	226	105	904	39	180	196	62	905	140	102	399	379	0													
12		3	235	77	846	42	164	176	67	738	101	121	362	327	0													
13		1	194	99	842	38	164	176	86	751	103	127	454	284	0													
14		1	198	106	1008	47	188	225	108	745	87	117	410	252	0													
15		0	216	83	856	35	140	135	71	633	83	123	302	185	0													
16		0	103	44	703	32	88	107	71	663	63	80	222	126	0													
17		0	92	66	707	21	82	118	74	610	59	75	270	127	0													
18		0	54	58	637	30	132	98	82	486	31	74	172	99	0													
19		0	41	35	511	29	113	95	56	460	31	70	159	72	0													
20		0	38	29	378	16	81	69	40	303	33	58	173	103	0													
21		0	34	18	298	19	88	44	25	263	22	56	167	96	0													
22		0	17	16	181	10	26	30	20	168	25	35	118	56	0													
23		0	33	9	146	3	23	33	8	111	20	20	106	36	0													
24		0	46	10	193	11	21	41	8	113	19	24	94	27	0													
25		0	26	11	133	7	15	14	7	71	10	13	68	38	0													
26		0	19	6	129	12	16	10	5	66	14	5	65	23	0													
27		0	17	13	73	12	7	7	3	60	8	3	38	12	0													
28		0	12	5	55	8	10	12	5	67	8	5	24	12	0													
29		0	5	5	52	4	8	8	5	50	4	3	17	8	0													
30		0	6	8	63	3	6	10	3	33	1	5	17	10	0													
31		0	11	5	44	1	0	9	3	25	2	2	22	7	0													

TABLE B-2 (Continued)

LOS &gt; 9

LOS	1622	CTA	1666	CTI	1633	CTM	1644	CTO	1655	CTR	1611	CTT	5080	CU	8300	DN	2100	DK	3200	DM	1900	DP	1010	DS	4100	EM	3800	EN
10		54		62		145		92		75		111		122		109		78		23		164		148		581		253
11		54		63		121		83		87		81		71		95		87		14		121		166		607		234
12		47		37		106		48		50		69		91		72		107		20		119		139		539		230
13		41		21		61		65		39		34		77		98		110		16		101		119		477		214
14		37		24		26		44		30		32		71		47		56		12		94		101		417		174
15		30		23		31		47		47		49		106		65		41		12		69		58		316		160
16		28		27		26		49		47		42		105		67		25		10		50		40		287		122
17		32		31		48		43		50		28		91		54		72		14		54		45		273		148
18		13		14		49		41		27		40		79		58		90		7		52		40		287		149
19		19		21		53		26		19		37		64		38		89		3		44		36		235		126
20		12		30		48		24		43		47		88		39		78		7		59		28		207		141
21		12		8		11		22		53		31		59		18		65		4		44		22		152		102
22		10		6		16		16		24		18		39		9		24		5		33		9		97		73
23		5		4		12		9		21		17		31		12		7		5		28		9		77		41
24		2		5		9		6		11		17		23		10		11		2		18		5		48		42
25		3		3		5		10		4		15		8		9		8		2		13		2		46		27
26		3		4		1		3		4		4		5		12		5		0		7		1		39		25
27		2		3		2		4		9		3		4		7		10		2		5		0		21		9
28		5		4		1		3		3		1		7		3		7		2		5		0		18		11
29		1		1		1		4		6		1		4		3		4		2		1		1		17		10
30		1		1		1		3		1		4		3		5		1		0		0		0		13		5
31		1		0		0		0		0		2		2		0		3		0		1		0		8		11

TABLE B-2 (Continued)

LOS &gt; 9

LOS	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
10	121	688	81	3	446	339	109	735	371	47	36	20	32	87
11	49	577	53	0	399	285	81	676	307	38	25	15	22	74
12	58	447	38	1	376	212	65	580	329	32	27	19	23	75
13	66	291	46	1	230	246	67	588	326	35	38	20	22	97
14	69	258	38	0	195	268	52	447	324	35	30	17	22	113
15	73	264	48	0	208	231	64	474	279	30	22	10	19	115
16	45	288	32	0	179	152	38	512	202	29	28	8	20	122
17	61	276	31	0	183	154	34	552	211	28	13	13	15	114
18	68	279	26	0	170	107	50	407	155	9	17	14	24	103
19	63	248	25	1	149	113	31	243	153	19	17	4	10	87
20	128	235	28	0	102	99	28	247	144	13	8	8	9	85
21	61	165	25	0	88	104	24	131	116	7	11	5	11	99
22	38	104	21	0	78	54	12	106	54	6	1	3	11	47
23	34	73	17	0	79	41	3	85	53	8	9	2	6	31
24	14	60	13	0	47	41	2	102	42	7	7	4	6	28
25	10	36	6	0	30	47	2	86	28	3	5	3	6	20
26	4	33	5	0	19	19	3	77	17	7	1	2	5	8
27	4	22	3	0	8	18	3	51	16	2	2	0	3	7
28	1	10	2	0	12	15	0	31	6	3	2	1	1	7
29	4	8	1	0	7	16	1	29	17	1	0	0	1	6
30	1	8	1	0	7	10	0	25	6	1	0	0	1	2
31	0	4	0	0	8	9	0	17	11	0	1	0	1	5

TABLE B-2 (Continued)

LOS &gt; 9

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	14600 PM	1800 PN
10	3	901	24	87	616	37	74	303	62	42	64	26	2	306
11	6	818	23	97	479	16	81	288	74	32	34	27	6	318
12	9	689	20	100	385	18	75	200	41	36	46	34	3	288
13	5	591	14	89	344	38	100	222	48	22	42	38	1	302
14	8	631	23	68	315	36	112	222	43	39	35	38	5	221
15	5	467	38	83	236	36	121	146	51	39	25	35	2	151
16	5	399	13	44	373	29	111	116	29	19	36	33	6	168
17	7	277	14	46	476	23	143	114	29	20	43	24	7	143
18	2	241	9	76	932	27	98	95	34	19	37	33	5	136
19	5	180	12	39	560	23	82	68	30	8	40	17	6	105
20	4	249	6	49	640	23	76	57	28	13	32	16	1	99
21	3	182	5	27	663	10	85	59	17	14	13	13	2	82
22	1	134	2	14	228	8	52	47	16	15	12	6	2	52
23	1	109	5	4	140	6	32	25	8	7	8	3	0	53
24	0	84	3	9	77	9	57	29	7	9	10	2	0	46
25	1	43	1	6	115	8	44	32	5	9	6	3	0	37
26	2	38	5	3	92	3	32	21	3	5	9	3	2	29
27	2	20	0	3	55	12	19	13	3	1	7	4	0	30
28	1	22	2	4	39	3	15	21	5	3	6	2	0	15
29	1	24	1	1	13	3	17	10	2	0	4	1	0	13
30	0	7	3	3	13	5	7	9	2	0	7	0	0	16
31	0	9	6	2	12	0	8	5	0	3	2	0	1	20

TABLE B-2 (Continued)

LOS &gt; 9

LOS	7000	PR	0200	QM	1500	RM	2500	RP	3600	SN	2490	SH	2000	SK	0250	SM	0400	ST	7200	TD	0500	TM	5800	UT	1700	YN	TOTAL
10		79	128		601		24		17		138		359		113		308		32		199			87		439	14252
11		78	116		547		18		6		213		483		92		228		27		166			57		458	12912
12		72	96		376		30		3		180		400		79		143		19		115			69		422	11163
13		73	114		419		19		5		183		403		87		142		27		78			65		486	10822
14		67	115		391		28		6		181		323		71		114		24		128			59		440	10268
15		63	113		412		26		1		169		215		70		105		25		102			70		384	8943
16		41	77		363		23		1		149		175		31		92		32		66			81		331	7725
17		42	79		326		25		1		164		241		46		84		40		74			52		282	7795
18		40	50		207		23		0		192		221		37		93		22		82			57		230	7326
19		28	31		179		13		0		94		220		23		78		25		52			38		188	5789
20		28	69		210		14		0		97		205		26		64		26		62			66		178	5644
21		14	33		190		9		0		94		217		33		53		12		22			51		125	4608
22		17	40		153		3		0		37		106		26		50		8		35			33		92	2805
23		7	40		109		3		0		26		73		18		30		5		31			15		73	2098
24		16	25		103		2		0		23		57		12		23		5		24			12		67	1900
25		12	31		68		3		1		19		37		15		15		2		12			12		65	1452
26		4	13		64		3		0		11		38		22		16		3		15			1		49	1169
27		8	12		41		0		0		12		21		12		7		1		8			4		29	797
28		5	4		41		0		0		5		40		6		7		4		11			2		23	670
29		4	6		37		1		0		8		18		10		2		2		6			2		30	532
30		4	6		22		0		0		11		23		8		2		3		6			6		15	443
31		3	14		15		1		0		8		19		5		4		1		9			0		23	385

TABLE B-3

FY 1986 WITHIN RATING TRANSITION RATES BY  
ENLISTMENT PROGRAM, RATING AND LOS

## 4YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.2795	0.3814	0.0975	0.2154	0.1905	0.2143	0.3333	4400 GS	0.0000	0.9167	0.7136	0.9344	0.5571	0.9815	0.9211
6700 AB	0.7108	0.9268	0.4314	0.7681	0.8333	0.8667	0.8261	8000 HM	0.7358	0.9334	0.5165	0.7767	0.8445	0.9037	0.8512
6600 AC	0.0000	0.9143	0.8113	0.5268	0.8824	0.9247	0.8376	4300 HT	0.7700	0.9350	0.3717	0.7350	0.9013	0.9593	0.8765
6080 AF	0.7865	0.9492	0.5258	0.8565	0.8983	0.8690	0.8741	2300 IS	0.6296	0.9597	0.6000	0.7358	0.8913	0.8462	0.8600
7100 AG	0.7273	0.9550	0.4818	0.8267	0.8636	0.8113	0.8537	2600 JO	1.0000	0.8421	0.9200	0.5667	0.8696	0.9333	0.8750
7300 AK	0.7994	0.9124	0.6953	0.8649	0.8975	0.8426	0.8333	3100 LI	0.0000	0.9200	0.6296	0.8378	0.7576	0.8824	0.8571
6500 AO	0.7752	0.9057	0.5206	0.7901	0.8348	0.9072	0.8830	1750 LN	0.0000	1.0000	0.7143	0.7391	0.8462	0.7333	0.8974
7500 AS	0.6849	0.9667	0.4771	0.8438	0.8939	0.8333	0.8594	0150 MA	0.0000	1.0000	1.0000	0.8182	0.9444	0.9583	0.9643
6180 AV	0.6814	0.9393	0.4468	0.8132	0.8804	0.8829	0.8158	4700 ML	0.7059	1.0000	0.6316	0.8182	1.0000	1.0000	1.0000
6400 AW	0.4810	0.9481	0.5113	0.7904	0.9381	0.9474	0.9111	3700 MM	0.6627	0.9180	0.4373	0.8159	0.9104	0.9350	0.8290
7400 AZ	0.7423	0.9415	0.6159	0.8718	0.8710	0.8323	0.8966	0900 MN	0.5735	0.9474	0.6000	0.8095	0.8095	0.8571	0.9565
0100 BM	1.0000	0.9386	0.4645	0.7460	0.8622	0.9173	0.8768	3900 MR	0.6932	0.9220	0.3167	0.8288	0.9083	0.9518	0.9455
4000 BT	0.5693	0.9332	0.4055	0.7488	0.8829	0.9475	0.8312	2200 MS	0.6706	0.8906	0.5262	0.8389	0.8900	0.9244	0.8555
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	1.0000	0.7125	0.8163	0.8235	0.9231	1.0000
1622 CTA	0.7191	0.9589	0.7912	0.8551	0.8696	0.9574	0.9796	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	1.0000
1666 CTH	0.9643	0.9773	0.6714	0.7234	0.9231	0.9655	0.9429	0300 OS	0.7348	0.9115	0.4187	0.7218	0.9276	0.8889	0.8304
1633 CTM	1.0000	1.0000	0.9565	0.9091	0.5385	0.8750	1.0000	0450 OT	0.7079	0.9060	0.6707	0.7176	0.8115	0.8750	0.9184
1644 CTO	0.6387	0.9333	0.6085	0.8476	0.8953	0.9286	0.7813	2700 PC	0.6176	0.9412	0.4186	0.7183	0.7963	0.8421	0.8966
1655 CTR	0.3354	0.9487	0.7561	0.7237	0.8750	0.8873	0.9355	7600 PH	0.0000	0.9487	1.0000	0.5600	0.6000	0.7391	0.8696
1611 CTT	0.9231	0.9476	0.7676	0.8056	0.9615	0.9027	0.9032	1080 PI	0.6846	0.9274	0.5385	0.8356	0.9268	1.0000	0.9091
5080 CU	0.0000	0.9053	0.8788	0.5000	0.7895	0.9286	0.8750	4600 PM	0.5500	1.0000	0.3889	0.8333	1.0000	1.0000	1.0000
8300 DN	1.0000	0.9518	0.9500	0.4699	0.6842	0.8810	0.9259	1800 PN	0.7003	0.9224	0.6419	0.8304	0.9141	0.8907	0.8877
2100 DK	0.6977	0.9304	0.7116	0.8566	0.8810	0.9160	0.8947	7000 PR	0.7586	0.9180	0.7063	0.8807	0.8571	0.8046	0.8727
3200 DM	0.0000	0.8571	0.6190	0.7826	1.0000	0.9333	0.8636	0200 QM	0.7373	0.8805	0.4809	0.8652	0.9669	0.9118	0.8537
1900 DP	0.0000	0.9407	0.9032	0.6053	0.7320	0.9000	0.8222	1500 RM	0.6917	0.9146	0.5133	0.8063	0.8730	0.9363	0.8926
1010 DS	0.0000	0.6250	0.6667	0.9375	0.9231	0.7500	0.8000	2500 RP	0.5959	0.8800	0.5670	0.8462	0.8605	0.9750	1.0000
4100 EM	0.5521	0.9383	0.4029	0.8475	0.9298	0.9640	0.8874	3600 SN	0.2420	0.4084	0.1180	0.2247	0.2727	0.4545	0.0952
3800 EN	0.7208	0.9284	0.4200	0.8216	0.9215	0.9327	0.8360	2490 SH	0.6900	0.9132	0.4363	0.7635	0.8333	0.9149	0.9044
5380 EQ	0.0000	0.8929	0.9756	0.4615	0.7826	0.8824	0.9412	2000 SK	0.7569	0.9308	0.5894	0.8545	0.8237	0.8913	0.9087
1000 ET	1.0000	0.9159	0.8365	0.8733	0.6771	0.8333	0.8810	0250 SM	0.6000	0.8717	0.5257	0.7607	0.8810	0.9362	0.7794
0350 EW	0.6935	0.9294	0.4440	0.8812	0.7451	0.8333	0.8889	0400 ST	0.7407	0.9399	0.6553	0.9059	0.6791	0.9178	0.9254
5000 FN	0.2792	0.3269	0.0668	0.0822	0.1579	0.2222	0.5000	7200 TD	0.0000	0.1667	0.1016	0.2857	0.2941	0.3684	0.3333
0800 FT	0.0000	0.9606	0.6462	0.8457	0.7748	0.9194	0.9000	0500 TM	0.5566	0.9417	0.4742	0.7725	0.8984	0.9433	0.8794
0600 GM	0.5611	0.9437	0.5084	0.8310	0.9440	0.9223	0.8586	5800 UT	0.0000	0.9167	0.9032	0.6129	0.8095	0.9167	0.8462
								1700 YN	0.7504	0.9239	0.6491	0.8158	0.8880	0.8705	0.7896

TABLE B-3 (Continued)

## ACTIVE MARINER PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.4226	0.0659	0.1572	0.4167	0.0714	0.5000	0.3333	4400 GS	0.5000	1.0000	0.5556	0.8000	0.7778	1.0000	0.6667
6700 AB	0.0000	0.3351	0.7190	0.8554	0.8235	0.8400	0.8140	8000 HM	0.7315	0.5079	0.6738	0.8198	0.8557	0.8480	0.8846
6600 AC	0.5972	0.5733	0.8611	0.7097	0.7778	0.8235	0.8571	4300 HT	0.8063	0.4244	0.7206	0.8733	0.9037	0.8871	0.8630
6080 AF	0.8153	0.5632	0.7705	0.9025	0.8790	0.8439	0.9130	2300 IS	0.6667	0.4565	0.8462	0.9091	0.7857	0.7273	1.0000
7100 AG	0.7959	0.5893	0.9444	0.7857	0.8846	0.8333	0.8333	2600 JO	0.6250	0.4286	1.0000	0.5000	1.0000	0.9167	0.8571
7300 AK	0.8542	0.6418	0.8485	0.8356	0.9524	0.8226	0.9400	3100 LI	0.0000	1.0000	0.3333	0.7500	0.7500	1.0000	0.0000
6500 AO	0.7562	0.4725	0.6905	0.8203	0.9189	0.8873	0.8548	1750 LN	0.0000	0.0000	1.0000	0.5000	0.8000	0.6667	1.0000
7500 AS	0.8333	0.5625	0.8295	0.8500	0.9189	0.8108	0.8788	0150 MA	0.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0.8333
6180 AV	0.6850	0.5178	0.6800	0.9030	0.8694	0.8170	0.8090	4700 ML	0.8000	0.5000	0.6667	1.0000	1.0000	0.6667	1.0000
6400 AW	0.4479	0.6329	0.7213	0.8485	0.9706	0.9512	0.8615	3700 MM	0.6125	0.4574	0.7164	0.9278	0.9180	0.8251	0.8791
7400 AZ	0.6667	0.6310	0.8710	0.9565	0.8710	0.8800	0.9643	0900 MN	0.3793	0.7500	0.7692	0.8333	0.6667	1.0000	1.0000
0100 BA	0.0000	0.3333	0.7176	0.8516	0.9167	0.8963	0.9167	3900 MR	0.8036	0.4184	0.7231	0.8936	0.9032	0.9375	1.0000
4000 BT	0.5271	0.4040	0.6106	0.9146	0.9250	0.8618	0.8113	2200 MS	0.6178	0.5050	0.7663	0.8786	0.8611	0.8882	0.8889
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	1.0000	1.0000	0.0000	1.0000	0.0000	1.0000
1622 CTA	0.8095	0.8667	1.0000	1.0000	1.0000	1.0000	1.0000	1400 NC	0.0000	0.0000	1.0000	1.0000	1.0000	0.0000	1.0000
1666 CTI	1.0000	0.8333	1.0000	1.0000	1.0000	0.7500	1.0000	0300 OS	0.7557	0.3908	0.6641	0.8208	0.8090	0.8462	0.9231
1633 CTM	0.6667	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	0450 OT	0.6429	0.5263	0.7692	0.8421	0.8333	1.0000	0.8333
1644 CTO	0.6923	0.6364	1.0000	1.0000	0.7143	0.7500	1.0000	2700 PC	0.0000	0.4500	0.7333	0.8333	1.0000	0.8889	0.8333
1655 CTR	0.2609	0.7778	0.8621	0.8182	1.0000	1.0000	1.0000	7600 PH	0.6250	0.6364	0.6250	0.6667	0.7500	0.8235	0.9000
1611 CTT	0.0000	0.6154	0.7500	0.9091	0.9412	0.8571	0.6364	1080 PI	0.7391	0.4839	0.9375	1.0000	0.8000	1.0000	1.0000
5080 CU	0.8148	0.4051	0.7838	0.9630	1.0000	0.8182	0.8636	4600 PM	0.0000	0.0000	1.0000	0.0000	1.0000	1.0000	0.0000
8300 DN	0.7243	0.6486	0.8571	0.6667	0.5714	0.8750	0.8333	1800 PN	0.7500	0.5645	0.8090	0.8438	0.9315	0.8252	0.9052
2100 DK	0.5833	0.6735	0.8302	0.9268	0.8571	0.8636	1.0000	7000 PR	0.7568	0.7222	0.7209	0.9412	0.9412	0.7083	1.0000
3200 DM	0.0000	0.5714	1.0000	1.0000	1.0000	1.0000	1.0000	0200 QM	0.7228	0.3939	0.7600	0.8983	0.9697	0.9143	0.8636
1900 DP	0.8333	0.7073	0.8824	0.6452	0.8800	0.8421	0.9375	1500 RM	0.6847	0.5303	0.7563	0.9426	0.9192	0.8553	0.7778
1010 DS	0.4444	0.9286	0.3636	0.8824	1.0000	0.8824	0.9091	2500 RP	0.0000	0.5333	0.8571	1.0000	0.8333	0.7143	0.6000
4100 EM	0.7105	0.4701	0.7248	0.9085	0.9167	0.8532	0.8649	3600 SN	0.3949	0.1084	0.1725	0.1803	0.2581	0.1304	0.0000
3800 EN	0.6667	0.4582	0.7130	0.8702	0.9318	0.9189	0.9231	2490 SH	0.5556	0.3333	0.6563	0.8000	0.9143	0.8302	0.9524
5380 EQ	0.7700	0.3052	0.7361	0.7895	0.9259	0.8636	0.9600	2000 SK	0.6838	0.4898	0.7600	0.8507	0.8772	0.8588	0.9324
1000 ET	0.5266	0.4699	0.6389	0.8936	0.8750	0.8205	0.9667	0250 SM	0.5275	0.5238	0.6957	0.9200	0.8889	1.0000	0.8696
0350 EW	0.7500	0.2836	0.7241	0.8889	0.7143	1.0000	1.0000	0400 ST	0.6832	0.5780	0.8904	0.8525	0.8286	0.9677	1.0000
5000 FN	0.4504	0.0583	0.1786	0.2778	0.4444	0.1667	0.0000	7200 TD	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.6054	0.5191	0.7706	0.9474	0.7750	0.7826	1.0000	0500 TM	0.5315	0.4737	0.8286	0.8214	0.9333	0.8478	0.8929
0600 GM	0.6344	0.4844	0.8344	0.8381	0.9178	0.8421	0.8788	5800 UT	0.7778	0.2740	0.6923	0.8500	0.9412	0.6667	1.0000
								1700 YN	0.7471	0.6525	0.8231	0.8788	0.8442	0.8485	0.8797



TABLE B-3 (Continued)

## 526YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	1.0000	0.4167	0.1864	0.0000	0.3636	0.0000	0.0000	4400 GS	0.6188	0.9433	0.7619	0.8955	0.5055	0.9231	0.8723
6700 AB	0.0000	0.8750	0.6538	0.7742	0.7000	0.7143	1.0000	8000 HM	0.8716	0.9537	0.8190	0.8497	0.6000	0.8497	0.8879
6600 AC	0.5358	0.9294	0.9424	0.4450	0.8016	0.9302	0.8889	4300 HT	0.8033	0.9167	0.7786	0.8936	0.6667	0.9444	0.9412
6080 AF	0.7500	0.8519	0.6849	0.8194	0.9024	0.8654	0.8621	2300 IS	1.0000	1.0000	0.6364	0.8571	1.0000	0.8750	1.0000
7100 AG	1.0000	1.0000	0.4000	0.8182	0.6000	0.8000	1.0000	2600 JO	0.5699	0.9000	0.9487	0.6757	0.7143	0.8696	0.6957
7300 AK	0.0000	1.0000	0.5556	0.8000	0.9091	0.9130	0.6667	3100 LI	0.0000	0.0000	0.5000	0.5000	1.0000	1.0000	0.0000
6500 AO	1.0000	0.8333	0.7000	0.8947	1.0000	0.7917	1.0000	1750 LN	0.0000	0.0000	0.5000	1.0000	1.0000	1.0000	1.0000
7500 AS	0.0000	0.7692	0.8462	0.6667	1.0000	1.0000	0.6667	0150 MA	0.0000	1.0000	0.0000	1.0000	1.0000	1.0000	1.0000
6180 AV	0.7508	0.9570	0.7471	0.8955	0.6868	0.8535	0.7696	4700 ML	0.0000	1.0000	0.5000	1.0000	1.0000	0.0000	0.0000
6400 AW	1.0000	0.9091	0.6111	0.7143	0.9000	0.8571	0.8571	3700 MM	0.8470	0.9341	0.9052	0.8304	0.5672	0.8944	0.8113
7400 AZ	1.0000	0.9286	0.6429	0.7308	0.9286	0.5000	1.0000	0900 MN	1.0000	1.0000	0.5000	1.0000	1.0000	0.0000	0.0000
0100 BM	0.9688	0.8476	0.4034	0.7059	0.7544	0.6786	0.8571	3900 MR	1.0000	0.8820	0.6071	0.8333	1.0000	1.0000	0.8333
4000 BT	0.7354	0.9035	0.7940	0.7415	0.5449	0.8947	0.9583	2200 MS	1.0000	0.8820	0.8374	0.7556	0.8377	0.9091	0.8571
6000 CN	0.0000	0.6000	0.0000	0.0000	0.0000	0.0000	1.0000	3300 MU	1.0000	1.0000	1.0000	0.8000	1.0000	0.0000	1.0000
1622 CTA	0.0000	0.8750	0.9231	0.8571	1.0000	1.0000	1.0000	1400 NV	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	1.0000
1666 CTH	0.0000	0.9773	0.8889	0.6250	0.9118	1.0000	1.0000	0300 OS	0.9524	0.8699	0.8873	0.5436	0.8839	0.9512	1.0000
1633 CTH	0.5920	0.9660	0.9624	0.9756	0.6741	0.8814	0.9010	0450 OT	0.6275	0.9130	0.8065	0.6190	0.7647	1.0000	0.7143
1644 CTR	0.0000	0.9600	0.6111	0.7778	0.9091	0.9333	0.6667	2700 PC	1.0000	0.9091	0.5385	0.8333	1.0000	1.0000	0.0000
1655 CTR	1.0000	0.9344	0.9464	0.8085	0.9375	1.0000	0.7500	7600 PH	0.5548	0.9406	0.9333	0.4069	0.7634	0.7805	0.8364
1611 CTT	0.0000	0.9815	0.9213	0.8293	0.9375	0.8333	0.7500	1080 PI	1.0000	0.9474	0.4286	1.0000	0.8000	1.0000	0.5000
5080 CU	0.7963	0.9028	0.8925	0.6074	0.8491	0.8667	0.9270	4600 PM	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	1.0000
8300 DN	0.6779	0.9204	0.9223	1.0000	0.7087	0.8750	0.8903	1800 PN	0.9286	0.9348	0.7143	0.8431	0.8824	0.9545	0.8462
2100 DK	1.0000	1.0000	0.7647	0.6000	0.8000	0.8750	1.0000	7000 PR	0.0000	0.9091	0.8889	0.8235	0.7500	1.0000	1.0000
3200 DM	0.0000	1.0000	1.0000	0.0000	0.0000	1.0000	0.8889	0200 QM	1.0000	0.9388	0.4146	0.9000	0.8500	0.9375	0.9091
1900 DP	0.8883	0.9295	0.9191	0.5413	0.7197	0.8112	0.8737	1500 RM	0.7778	0.9397	0.8128	0.8632	0.6000	0.9099	0.9070
1010 DS	0.6085	0.9345	0.8491	0.9234	0.4850	0.6860	0.8696	2500 RP	1.0000	0.8333	0.5000	0.7778	0.6000	0.6000	1.0000
4100 EM	0.8968	0.9439	0.8430	0.8644	0.5803	0.9219	0.8344	3600 SN	0.7826	0.3461	0.1322	0.2154	0.0500	0.2857	0.6000
3800 EN	1.0000	0.9706	0.4906	0.7907	0.8667	0.9677	0.9091	2490 SH	1.0000	0.9091	0.3750	0.6316	0.7273	0.8182	0.8750
5380 EQ	0.7638	0.9170	0.9061	0.4988	0.7970	0.8682	0.9697	2000 SK	1.0000	0.9515	0.6604	0.8333	0.8108	0.9565	1.0000
1000 ET	0.6936	0.9496	0.9393	0.9394	0.5734	0.8491	0.8712	0250 SM	1.0000	0.9038	0.7656	0.6667	0.8182	1.0000	1.0000
0350 EW	0.7610	0.9502	0.7933	0.9312	0.5333	0.8491	0.9211	0400 ST	0.7251	0.9392	0.9174	0.9395	0.5806	0.9110	0.9344
5000 FN	0.7966	0.2532	0.0986	0.0000	0.0000	0.0000	0.5000	7200 TD	0.0000	0.0000	0.0000	0.3333	0.0000	1.0000	0.5000
0800 FT	0.7293	0.9443	0.8957	0.9180	0.5920	0.9101	0.8854	0500 TM	1.0000	1.0000	0.8069	0.6019	0.8837	0.9600	1.0000
0600 GM	1.0000	0.9524	0.7951	0.6869	0.8980	0.9583	0.8667	5800 UT	1.0000	0.9231	0.9091	0.5077	0.8219	0.7475	0.9091
								1700 YN	1.0000	0.7778	0.6667	0.8116	0.9063	0.9048	0.6842

TABLE B-3 (Continued)

## PRIOR SERVICE PROGRAM

Rate	1-3	1-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0054	0.3000	0.2439	0.2632	0.1250	0.1875	0.0000	4400 GS	0.6667	1.0000	1.0000	0.7778	0.9444	0.9412	0.9048
6700 AB	0.7500	1.0000	0.8800	0.7241	0.8649	0.8133	0.8936	8000 HM	0.8322	0.9444	0.8519	0.7891	0.8440	0.9031	0.8580
6600 AC	0.7857	0.0000	1.0000	1.0000	0.9545	0.8780	0.7719	4300 HT	0.8210	0.8750	0.7500	0.8571	0.8669	0.8700	0.8466
6080 AF	0.3529	0.7857	0.8182	0.8457	0.8636	0.8835	0.8915	2300 IS	1.0000	1.0000	1.0000	1.0000	0.9000	0.9130	0.9091
7100 AG	1.0000	0.0000	0.5000	0.5000	0.7500	0.9130	0.7750	2600 JO	0.0000	0.0000	0.0000	1.0000	0.8571	0.8571	0.9091
7300 AK	0.7143	1.0000	1.0000	0.7381	0.9333	0.8767	0.9054	3100 LI	0.6667	0.0000	0.0000	0.8333	0.7500	0.7647	0.9231
6500 AO	0.7882	0.8889	0.8043	0.8737	0.8921	0.8782	0.8786	1750 LN	0.0000	0.0000	1.0000	1.0000	0.6667	1.0000	0.9583
7500 AS	0.9545	1.0000	1.0000	0.6818	0.8750	0.8919	0.9259	0150 MA	1.0000	1.0000	1.0000	1.0000	1.0000	0.9286	1.0000
6180 AV	0.8488	0.8824	0.7119	0.7500	0.8701	0.8552	0.8635	4700 ML	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	1.0000
6400 AW	0.6667	0.8750	0.9412	0.8889	0.9429	0.8125	0.9130	3700 MM	0.8571	0.7692	0.7547	0.7829	0.8671	0.8990	0.8585
7400 AZ	0.9375	1.0000	0.7778	0.8571	0.8605	0.8600	0.8714	0900 MN	1.0000	1.0000	0.5000	1.0000	1.0000	0.6364	0.9091
0100 BM	0.7500	1.0000	0.6500	0.8217	0.7905	0.8553	0.8419	3900 MR	0.9545	1.0000	0.9231	0.9487	0.8983	0.9412	0.9245
4000 BT	0.7229	1.0000	0.7241	0.8571	0.7890	0.8226	0.7707	2200 MS	0.7442	0.8182	0.9057	0.8276	0.7895	0.8679	0.8484
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	3300 MU	1.0000	0.0000	1.0000	0.8750	1.0000	0.8571	0.6923
1622 CTA	0.0000	0.0000	0.0000	1.0000	0.5000	1.0000	0.7778	1400 NC	1.0000	0.0000	0.0000	0.0000	1.0000	1.0000	1.0000
1666 CTI	1.0000	1.0000	1.0000	1.0000	1.0000	0.8000	0.8696	0300 OS	0.8421	0.8333	0.7209	0.8679	0.8966	0.8933	0.8685
1633 CTM	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.8824	0450 OT	0.7000	0.0000	0.8000	0.8333	0.9524	0.8077	0.9375
1644 CTO	0.5000	1.0000	0.5000	0.6667	0.9231	0.7500	0.8727	2700 PC	0.6154	1.0000	1.0000	1.0000	0.7917	0.8108	0.8929
1655 CTR	0.2857	0.0000	1.0000	0.5714	0.8333	0.9630	0.9184	7600 PH	0.0000	0.0000	1.0000	0.0000	0.6667	0.6842	0.9231
1611 CTT	0.8571	0.0000	1.0000	0.7143	0.8235	0.8846	0.9302	1080 PI	0.8333	0.0000	0.7500	0.8571	1.0000	0.8636	0.9286
5080 CU	0.9200	0.7778	0.7692	0.7619	0.9444	0.8254	0.8889	4600 PM	1.0000	1.0000	0.0000	1.0000	1.0000	1.0000	0.8750
8300 DN	1.0000	0.0000	0.6667	0.7692	0.7826	0.8750	0.9444	1800 PN	0.6250	0.6667	0.6875	0.7000	0.8333	0.8730	0.8372
2100 DK	0.8462	0.6667	1.0000	0.7692	0.8788	0.7857	0.7317	7000 PR	0.7273	1.0000	0.7778	0.8750	0.9286	0.9474	0.8966
3200 DM	1.0000	0.0000	0.0000	0.5000	0.8571	0.8000	1.0000	0200 QM	0.7907	0.0000	0.7407	0.8590	0.9014	0.9111	0.8571
1900 DP	1.0000	0.5000	1.0000	0.8889	0.7576	0.9077	0.8947	1500 RM	0.7640	0.8095	0.6494	0.8578	0.8789	0.8731	0.8810
1010 DS	0.5000	0.0000	0.6667	1.0000	0.8000	0.7500	0.8696	2500 RP	0.9091	1.0000	1.0000	0.9500	0.9167	0.8333	0.8125
4100 EM	0.9364	1.0000	0.8679	0.8514	0.9364	0.8924	0.8425	3600 SN	0.0057	0.4054	0.2245	0.2649	0.1818	0.2653	0.2593
3800 EN	0.8452	0.7778	0.8000	0.9028	0.8267	0.8936	0.8784	2400 SH	0.7895	0.3333	0.3333	0.5789	0.7969	0.8125	0.7911
5380 EQ	0.8182	0.6667	0.7500	0.7857	0.9231	0.9375	0.9444	2000 SK	0.8406	0.8333	0.7931	0.8226	0.8621	0.8921	0.9044
1000 ET	0.8723	1.0000	0.9000	0.9091	0.8889	0.9355	0.9173	0250 SM	0.7317	1.0000	0.9333	0.8833	0.8500	0.8475	0.9059
0350 EW	0.9000	1.0000	0.8333	1.0000	1.0000	0.9333	0.8000	0400 ST	0.8605	0.8333	0.8387	0.9796	0.9394	0.9706	0.8289
5000 FN	0.0019	0.1667	0.2174	0.2667	0.1250	0.1250	0.0000	7200 TD	0.0000	0.0000	0.5000	0.0000	0.0000	0.3333	0.0000
0800 FT	0.8696	0.7500	0.9524	0.8718	0.9286	0.9535	0.9138	0500 TM	0.7000	1.0000	0.3636	0.6000	0.8929	0.8776	0.7927
0600 GM	0.7609	0.5000	0.9149	0.9449	0.9535	0.8970	0.8538	5800 UT	0.9231	0.8571	0.8000	0.7778	1.0000	0.9167	0.9231
								1700 YN	0.5556	0.7500	0.8235	0.7925	0.8125	0.7664	0.8917

TABLE B-3 (Continued)

## TAR PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0000	0.4706	0.2500	0.0000	0.0000	0.0000	0.0000	4400 GS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6700 AB	0.0000	0.0000	1.0000	1.0000	1.0000	0.0000	0.0000	8000 HM	0.7463	1.0000	1.0000	0.8846	1.0000	1.0000	0.9286
6000 AC	0.6154	1.0000	0.5556	1.0000	1.0000	1.0000	0.0000	4300 HT	0.8478	1.0000	0.8333	0.9565	0.8750	1.0000	1.0000
6000 AF	0.8800	0.9500	0.5899	0.9000	0.9043	0.9405	0.8627	2300 IS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7100 AG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2600 JO	0.6667	1.0000	0.5000	1.0000	0.0000	1.0000	1.0000
7300 AK	0.7403	0.9474	0.5000	0.8846	0.8000	0.8750	1.0000	3100 LI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6500 AO	0.6842	1.0000	0.7600	0.8571	0.8000	1.0000	1.0000	1750 LN	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
7500 AS	0.7619	1.0000	0.7500	0.6667	1.0000	1.0000	0.8000	0150 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6100 AV	0.7294	0.9651	0.6815	0.8404	0.8293	0.9286	0.9091	4700 ML	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
6400 AW	0.6000	0.9412	0.3636	0.9412	0.7000	1.0000	0.8333	3700 MM	0.3824	0.9000	0.5833	0.9655	0.9500	0.8095	1.0000
7400 AZ	0.8000	0.8333	0.8000	0.7273	0.8571	1.0000	0.6667	0900 MN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0000	1.0000	0.6667	0.9412	0.9524	1.0000	1.0000	3900 MR	0.8182	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
4000 BT	0.6667	0.9444	0.7778	1.0000	0.9375	0.9375	0.6364	2200 MS	0.0000	0.0000	0.0000	0.0000	1.0000	1.0000	0.0000
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTI	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0300 OS	0.0000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1644 CTO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1655 CTR	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7600 PH	0.0000	1.0000	0.0000	0.0000	0.0000	1.0000	0.0000
1611 CTT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1080 PI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5080 CU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1800 PN	0.6078	1.0000	0.8125	0.8710	0.9643	0.9130	0.9615
2100 DK	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	7000 PR	0.6897	0.9167	0.5000	0.7273	1.0000	1.0000	0.7500
3200 DM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0200 QM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1900 DP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	1500 RM	0.0000	1.0000	0.0000	0.0000	0.0000	0.6667	1.0000
1010 DS	0.4000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2500 RP	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
3800 EN	0.6667	0.8000	0.6667	0.8889	0.9333	1.0000	0.8333	3600 SH	0.0000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000
5300 EQ	0.0000	0.0000	0.0000	0.9286	1.0000	1.0000	1.0000	2490 SN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1000 ET	0.5250	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2000 SK	0.6923	1.0000	0.8125	0.9118	0.9524	0.9355	0.9091
0350 EW	0.0000	0.0000	0.0000	0.6667	1.0000	0.8333	1.0000	0250 SM	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
5000 FN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0400 ST	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0600 GM	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0500 TM	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
				0.0000	0.0000	0.0000	0.0000	5800 UT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
								1700 YN	0.7480	1.0000	0.8519	0.9265	0.9808	1.0000	0.9286

TABLE B-3 (Continued)

LOS &gt; 9

	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
9-10	0.0000	0.8864	0.9100	0.9372	0.9318	0.9363	0.9102	0.9259	0.8835	0.9157	0.9625	0.8879	0.8339	0.0000
10-11	0.5000	0.9055	0.8421	0.9305	0.9296	0.9312	0.9195	0.9535	0.8684	0.8675	0.8803	0.9145	0.8940	0.0000
11-12	0.0000	0.9690	0.9143	0.9524	0.8462	0.9389	0.9388	0.9516	0.9348	0.9214	0.9412	0.9323	0.9261	0.0000
12-13	0.3333	0.9277	0.9091	0.9563	0.9286	0.9573	0.9602	0.9254	0.9322	0.9208	0.9256	0.9199	0.9419	0.0000
13-14	0.0000	0.9588	0.8990	0.9727	0.8947	0.9695	0.9375	1.0000	0.9441	0.9126	0.9370	0.9427	0.9261	0.0000
14-15	0.0000	0.9394	0.9245	0.9692	0.9574	0.9734	0.9378	0.9630	0.9624	0.9655	0.9915	0.9634	0.9444	0.0000
15-16	0.0000	0.9676	0.9518	0.9708	0.9429	0.9643	0.9185	0.9437	0.9779	0.9639	0.9431	0.9470	0.9676	0.0000
16-17	0.0000	0.9806	0.9545	0.9872	1.0000	0.9659	0.9626	1.0000	0.9744	0.9206	0.9875	0.9459	0.9683	0.0000
17-18	0.0000	0.9674	0.9697	0.9887	0.9524	1.0000	0.9407	0.9865	0.9820	0.9661	0.9733	0.9852	0.9843	0.0000
18-19	0.0000	0.9444	0.9828	0.9890	1.0000	1.0000	0.9592	0.9878	0.9918	1.0000	0.9865	0.9826	0.9596	0.0000
19-20	0.0000	0.9268	0.9429	0.9530	0.9655	0.9646	0.9789	0.9107	0.9543	0.9032	0.9429	0.9371	0.9722	0.0000
20-21	0.0000	0.6579	0.5862	0.6429	0.5625	0.6543	0.5217	0.5500	0.6337	0.5455	0.6379	0.6243	0.7292	0.0000
21-22	0.0000	0.6471	0.6667	0.7550	0.6842	0.6591	0.8409	0.6800	0.7034	0.6364	0.7500	0.7725	0.6214	0.0000
22-23	0.0000	0.6471	0.8125	0.7514	1.0000	0.8077	0.5667	0.6500	0.6845	0.5200	0.6857	0.7627	0.7857	0.0000
23-24	0.0000	0.6970	0.6667	0.7808	0.6667	0.7391	0.7273	0.8750	0.7658	0.5200	0.8500	0.8113	0.7778	0.0000
24-25	0.0000	0.8913	0.9000	0.8446	0.6364	0.6190	0.8780	0.5000	0.7876	0.5789	0.9167	0.8191	0.7037	0.0000
25-26	0.0000	0.9231	1.0000	0.8421	0.8571	0.7333	1.0000	0.7143	0.8873	0.9000	0.7692	0.8971	0.7105	0.0000
26-27	0.0000	0.8421	0.8333	0.8915	0.8333	0.7500	0.7000	0.8000	0.7879	0.5714	0.6000	0.8308	0.8261	0.0000
27-28	0.0000	0.8235	0.6923	0.8082	0.6667	0.7143	1.0000	0.6667	0.7833	0.6250	0.6667	0.7895	0.8333	0.0000
28-29	0.0000	0.8333	0.8000	0.9273	0.8750	0.8000	0.9167	1.0000	0.8806	1.0000	1.0000	0.7917	0.9167	0.0000
29-30	0.0000	1.0000	0.8000	0.7500	1.0000	0.7500	0.5000	0.6000	0.7800	0.5000	0.6667	0.8824	0.6250	0.0000
30-31	0.0000	0.1667	0.5000	0.2381	0.3333	0.3333	0.3000	0.3333	0.3939	0.0000	0.6000	0.3529	0.7000	0.0000

TABLE B-3 (Continued)

LOS &gt; 9

	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
9-10	0.9600	0.9333	0.8889	0.9643	0.9298	0.9375	0.8837	0.9483	0.9180	0.8571	0.9200	0.9324	0.8898	0.9440
10-11	0.9074	0.9355	0.8276	0.9239	0.9600	0.9820	0.9098	0.9083	0.9487	0.8696	0.9085	0.7973	0.8898	0.9051
11-12	0.9074	0.9206	0.9339	0.9398	0.9885	1.0000	0.9437	0.9083	0.9540	0.9286	0.9339	0.8735	0.9423	0.9530
12-13	1.0000	0.9189	0.9057	0.9583	0.9400	0.9420	0.9560	0.9444	0.9813	1.0000	0.9076	0.8921	0.9165	0.9609
13-14	1.0000	0.8571	0.9344	0.9538	1.0000	0.9706	0.9740	0.9592	0.9727	1.0000	0.9109	0.8739	0.9350	0.9533
14-15	1.0000	0.9167	0.9615	0.9545	0.9333	0.9375	0.9577	0.9574	0.9464	0.8333	0.9468	0.9604	0.9329	0.9483
15-16	0.9667	0.9130	1.0000	0.9574	0.9574	0.8980	0.9717	0.9692	0.9756	1.0000	0.9710	0.9483	0.9715	0.9688
16-17	1.0000	0.9630	0.9615	0.9388	0.9362	0.9762	0.9905	0.9851	1.0000	0.9000	0.9400	0.9750	0.9756	0.9754
17-18	0.9375	1.0000	0.9583	0.9535	0.9400	1.0000	1.0000	1.0000	1.0000	1.0000	0.9828	0.9556	0.9927	0.9865
18-19	1.0000	1.0000	0.9796	0.9024	1.0000	0.9000	1.0000	0.9828	0.9889	1.0000	0.9423	0.9750	0.9895	0.9866
19-20	1.0000	1.0000	0.9623	0.8462	0.8947	0.8919	0.9219	0.9474	0.9663	0.6667	0.9545	0.9722	0.9532	0.9683
20-21	0.7500	0.8000	0.7292	0.7917	0.6744	0.7872	0.5909	0.6410	0.6795	0.2857	0.5593	0.6071	0.5749	0.6596
21-22	0.5000	0.6250	0.4545	0.5909	0.7547	0.8065	0.6441	0.8333	0.7231	0.5000	0.6364	0.6818	0.5526	0.5490
22-23	0.6000	0.8333	0.7500	0.8125	0.5417	0.7222	0.6923	0.3333	0.7083	0.6000	0.8485	1.0000	0.6701	0.6712
23-24	1.0000	0.5000	0.4167	0.6667	0.4762	0.8235	0.7419	1.0000	0.7143	0.6000	0.7857	0.4444	0.6623	0.8537
24-25	1.0000	0.8000	0.5556	0.5000	0.5455	0.7059	0.7826	0.9000	0.8182	1.0000	0.7778	1.0000	0.7917	0.7381
25-26	0.6667	1.0000	1.0000	0.7000	1.0000	0.9333	0.8750	0.7778	0.5000	0.5000	0.8462	1.0000	0.7391	0.7778
26-27	1.0000	0.7500	1.0000	0.6667	0.7500	0.5000	0.8000	0.8333	0.8000	0.0000	0.8571	1.0000	0.7692	0.8400
27-28	1.0000	1.0000	1.0000	0.5000	0.7778	0.6667	0.5000	0.4286	0.8000	0.0000	1.0000	1.0000	0.7619	0.5556
28-29	0.4000	1.0000	0.0000	0.6667	0.6667	1.0000	1.0000	1.0000	1.0000	0.5000	0.8000	0.0000	0.7778	1.0000
29-30	1.0000	1.0000	1.0000	1.0000	0.6667	1.0000	0.7500	1.0000	0.7500	1.0000	0.0000	1.0000	0.8235	0.8000
30-31	0.0000	0.0000	0.0000	0.3333	1.0000	0.2500	0.0000	0.2000	1.0000	0.0000	0.0000	0.0000	0.4615	0.8000

TABLE B-3 (Continued)

LOS &gt; 9

	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
9-10	0.9565	0.9091	1.0000	0.0000	0.8788	0.9348	0.9074	0.8894	0.8922	0.9545	0.9500	0.8947	0.8421	0.8519
10-11	0.9421	0.8387	0.8519	0.0000	0.8274	0.9263	0.8716	0.9007	0.9164	0.9149	0.9444	0.9500	0.9688	0.9425
11-12	1.0000	0.9012	0.9245	0.0000	0.8997	0.9228	0.9259	0.9142	0.8958	0.9211	0.9200	0.9333	0.8636	0.9459
12-13	0.9655	0.9172	0.9737	0.0000	0.9069	0.9292	0.9538	0.9224	0.9362	0.9688	0.9630	0.7895	0.9565	0.9600
13-14	0.9394	0.9278	0.9783	1.0000	0.9565	0.9431	0.9254	0.9439	0.9509	0.9429	0.9211	0.9500	0.9545	0.9381
14-15	0.9710	0.9767	0.8947	0.0000	0.9385	0.9478	1.0000	0.9463	0.9383	0.9429	0.9667	0.8824	0.9545	0.9646
15-16	0.9863	0.9545	0.9583	0.0000	0.9615	0.9610	0.9688	0.9768	0.9785	0.9000	0.9545	1.0000	0.9474	0.9652
16-17	0.9556	0.9549	0.9375	0.0000	0.9888	0.9211	0.9737	0.9727	0.9604	1.0000	1.0000	1.0000	0.9000	0.9918
17-18	1.0000	0.9710	1.0000	0.0000	0.9891	0.9610	0.9706	0.9819	0.9668	0.8929	1.0000	1.0000	1.0000	0.9649
18-19	0.9853	0.9857	0.9231	0.0000	0.9941	0.9720	0.9800	0.9828	0.9677	1.0000	1.0000	1.0000	1.0000	0.9903
19-20	0.9206	0.9597	0.9600	0.0000	0.9463	0.9558	0.9032	0.9342	0.9477	0.8947	0.8824	1.0000	0.9000	0.9540
20-21	0.5547	0.5702	0.6429	0.0000	0.7843	0.5556	0.6786	0.6316	0.6458	0.5385	0.6250	0.5000	0.5556	0.6824
21-22	0.7377	0.6970	0.8400	0.0000	0.6818	0.6923	0.7083	0.7023	0.7241	0.7143	0.8182	0.4000	0.6364	0.7172
22-23	0.6842	0.7115	0.7143	0.0000	0.7436	0.8333	0.6667	0.7453	0.7407	0.5000	1.0000	0.6667	0.9091	0.8723
23-24	0.6765	0.7671	0.8824	0.0000	0.7595	0.8293	0.6667	0.7882	0.8113	0.6250	0.6667	1.0000	0.6667	0.7097
24-25	0.7143	0.7167	0.9231	0.0000	0.8298	0.7317	1.0000	0.7647	0.9286	1.0000	0.8571	0.7500	0.6667	0.6786
25-26	0.8000	0.8056	1.0000	0.0000	0.8000	0.8085	0.5000	0.8605	0.8571	1.0000	0.8000	0.6667	0.8333	0.8000
26-27	0.5000	0.8182	0.8000	0.0000	0.7895	0.7368	0.6667	0.8182	0.8235	1.0000	1.0000	1.0000	0.6000	0.8750
27-28	0.7500	0.8182	1.0000	0.0000	1.0000	0.8333	1.0000	0.8235	0.7500	1.0000	1.0000	0.0000	0.6667	0.8571
28-29	1.0000	0.9000	0.5000	0.0000	1.0000	1.0000	0.0000	0.7419	0.8333	0.3333	0.5000	1.0000	1.0000	1.0000
29-30	1.0000	0.8750	1.0000	0.0000	1.0000	0.7500	1.0000	0.8621	1.0000	0.0000	0.0000	0.0000	1.0000	0.8333
30-31	0.0000	0.3750	0.0000	0.0000	0.2857	0.4000	0.0000	0.2800	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-3 (Continued)

LOS &gt; 9

	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
9-10	1.0000	0.9136	0.8824	0.9167	0.9091	1.0000	0.9500	0.8440	0.9063	0.8000	0.8750	0.9333	1.0000	0.8840
10-11	1.0000	0.8868	0.8750	0.8966	0.8896	0.9459	0.9189	0.8878	0.9194	0.9286	0.9531	0.8846	1.0000	0.9150
11-12	1.0000	0.9328	1.0000	0.9794	0.9499	0.9375	0.9506	0.9236	0.8514	0.9375	0.9412	0.8889	1.0000	0.9434
12-13	1.0000	0.9390	0.9000	0.9800	0.9429	1.0000	0.9467	0.8850	0.9512	0.9444	0.9783	0.9118	0.6667	0.9236
13-14	1.0000	0.9154	1.0000	0.9888	0.9593	0.9474	0.9600	0.9459	0.9792	0.9091	0.8333	0.9737	1.0000	0.9669
14-15	1.0000	0.9366	1.0000	0.9853	0.9524	0.9722	0.9375	0.9144	0.9535	0.9231	0.9429	0.9474	1.0000	0.9367
15-16	0.8000	0.9700	0.9211	0.9759	0.9703	0.9722	0.9504	0.9569	0.9020	1.0000	1.0000	1.0000	1.0000	0.9536
16-17	1.0000	0.9649	0.9231	0.9773	0.9893	0.9655	0.9550	0.9569	0.9655	1.0000	1.0000	1.0000	0.8333	0.9881
17-18	1.0000	0.9711	1.0000	0.9348	0.9958	1.0000	0.9720	0.9649	1.0000	1.0000	0.9535	1.0000	1.0000	0.9580
18-19	1.0000	0.9834	1.0000	1.0000	0.9936	1.0000	0.9490	0.9694	1.0000	1.0000	1.0000	1.0000	1.0000	0.9559
19-20	1.0000	0.9389	0.9167	0.9487	0.9679	0.9130	0.9024	0.8676	0.8667	0.8750	0.9750	0.9412	1.0000	0.9619
20-21	1.0000	0.6104	0.6667	0.5102	0.6766	0.6957	0.6447	0.5965	0.5000	0.8462	0.6875	0.5000	1.0000	0.7172
21-22	0.6667	0.7418	0.6000	0.5556	0.5807	0.9000	0.7765	0.6780	0.5294	0.5714	0.6923	0.5385	0.0000	0.7439
22-23	0.0000	0.7687	1.0000	0.4286	0.6579	0.7500	0.7115	0.7021	0.7500	1.0000	0.6667	0.6667	1.0000	0.6538
23-24	0.0000	0.6881	1.0000	0.7500	0.7429	0.8333	0.8125	0.7600	0.8750	0.7143	0.7500	1.0000	0.0000	0.6981
24-25	0.0000	0.7500	0.6667	0.7778	0.6623	0.8889	0.8772	0.7241	0.7143	0.7778	0.2000	0.5000	0.0000	0.8696
25-26	1.0000	0.7674	0.0000	1.0000	0.8174	1.0000	0.7955	0.7500	1.0000	0.8889	0.5000	0.6667	0.0000	0.7838
26-27	1.0000	0.9737	1.0000	0.6667	0.7500	0.6667	0.8438	0.7619	0.3333	1.0000	1.0000	0.6667	1.0000	0.7241
27-28	1.0000	0.8500	0.0000	0.6667	0.7636	0.6667	0.8947	0.9231	1.0000	0.0000	1.0000	0.7500	0.0000	0.8667
28-29	1.0000	0.9545	1.0000	1.0000	0.9487	0.6667	0.7333	0.9048	0.8000	1.0000	0.8333	1.0000	0.0000	0.8667
29-30	1.0000	0.9167	1.0000	1.0000	0.9231	1.0000	0.8235	0.9000	1.0000	0.0000	0.7500	1.0000	0.0000	0.3846
30-31	0.0000	0.4286	0.3333	0.3333	0.1538	0.2000	0.5714	0.2222	0.0000	0.0000	0.1429	0.0000	0.0000	0.3750

TABLE B-3 (Continued)

LOS &gt; 9

	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN
9-10	0.9474	0.8718	0.8870	1.0000	0.2857	0.8557	0.9474	0.8421	0.7872	0.2333	0.9026	0.9474	0.9000
10-11	0.9367	0.9219	0.8985	0.7500	0.1176	0.8986	0.9443	0.8230	0.8636	0.4063	0.8894	0.9080	0.9134
11-12	0.9487	0.9138	0.9360	0.8889	0.1667	0.9437	0.9689	0.9022	0.9254	0.2593	0.9699	0.9649	0.9432
12-13	0.9722	0.9583	0.9441	0.8333	0.0000	0.9611	0.9675	0.8608	0.9441	0.4211	0.9565	0.9565	0.9479
13-14	0.9178	0.9825	0.9570	0.9474	0.2000	0.9454	0.9578	0.9080	0.9296	0.3704	0.8974	0.9846	0.9383
14-15	0.9701	0.9391	0.9361	0.9643	0.5000	0.9337	0.9845	0.9437	0.9649	0.3333	0.9766	0.9492	0.9636
15-16	0.9841	0.9558	0.9636	1.0000	0.0000	0.9704	0.9767	0.9000	0.9714	0.3600	0.9412	0.9143	0.9714
16-17	0.9756	0.9610	0.9697	0.9565	0.0000	0.9799	0.9657	0.9355	0.9674	0.5313	0.9545	0.9630	0.9637
17-18	0.9762	0.9747	0.9755	0.9200	0.0000	0.9878	0.9668	0.9783	0.9643	0.6000	0.9595	0.9615	0.9574
18-19	0.9750	0.9600	0.9903	0.9565	0.0000	0.9896	0.9819	0.9459	1.0000	0.8182	0.9756	0.9649	0.9826
19-20	0.9643	0.9355	0.9777	0.9231	0.0000	0.9468	0.9773	0.9130	0.9359	0.8000	0.9038	0.8947	0.9468
20-21	0.6429	0.6667	0.5952	0.7857	0.0000	0.6804	0.6829	0.7308	0.6875	0.1923	0.5323	0.5303	0.6910
21-22	0.6429	0.6970	0.6789	0.6667	0.0000	0.6809	0.7512	0.8485	0.7736	0.4167	0.6364	0.6078	0.7440
22-23	0.6471	0.8750	0.7059	1.0000	0.0000	0.7027	0.6604	0.6154	0.7400	0.2500	0.6857	0.6364	0.7391
23-24	0.8571	0.8000	0.7156	0.6667	0.0000	0.6538	0.7123	0.9444	0.8000	0.2000	0.7097	0.5333	0.7260
24-25	0.7500	0.8800	0.7379	1.0000	0.0000	0.7391	0.7895	1.0000	0.7391	0.0000	0.7500	0.8333	0.7910
25-26	1.0000	0.7742	0.8824	0.6667	0.0000	1.0000	0.7838	0.8667	0.8667	1.0000	0.9167	0.5833	0.8615
26-27	0.7500	0.6923	0.8125	1.0000	0.0000	1.0000	0.7895	0.7727	0.7500	0.3333	0.8000	0.0000	0.7755
27-28	0.7500	0.8333	0.7561	0.0000	0.0000	0.8333	0.7143	0.9167	0.8571	1.0000	0.7500	0.2500	0.6897
28-29	0.8000	1.0000	0.8537	0.0000	0.0000	0.8000	0.8500	0.6667	1.0000	0.7500	0.8182	0.5000	0.8696
29-30	0.5000	0.6667	0.7838	1.0000	0.0000	0.8750	0.6667	0.9000	0.5000	0.0000	1.0000	0.5000	0.7333
30-31	0.7500	0.6667	0.4545	0.0000	0.0000	0.3636	0.3043	0.5000	0.5000	0.0000	0.8333	0.0000	0.2000



TABLE B-4

FY 1986 LATERAL TRANSFER RATES BY  
ENLISTMENT PROGRAM, RATING AND LOS

## 4YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0435	0.0064	0.0122	0.0067	0.0222	0.0128	0.0000	4400 GS	0.0024	0.0015	0.0027	0.0168	0.0074	0.0000	0.0000
6700 AB	0.0382	0.0451	0.0571	0.0503	0.0444	0.0128	0.0122	8000 HM	0.0383	0.0201	0.0109	0.0168	0.0148	0.0000	0.0244
6600 AC	0.0076	0.0030	0.0027	0.0101	0.0000	0.0256	0.0244	4300 HT	0.0410	0.0410	0.0190	0.0134	0.0296	0.0000	0.0244
6080 AF	0.0535	0.0501	0.0381	0.0268	0.0370	0.0256	0.0366	2300 IS	0.0026	0.0030	0.0041	0.0101	0.0074	0.0128	0.0000
7100 AG	0.0033	0.0008	0.0027	0.0000	0.0000	0.0000	0.0000	2600 JO	0.0021	0.0027	0.0041	0.0034	0.0000	0.0128	0.0000
7300 AK	0.0116	0.0197	0.0150	0.0134	0.0222	0.0385	0.0122	3100 LI	0.0025	0.0038	0.0122	0.0101	0.0074	0.0000	0.0000
6500 AO	0.0185	0.0148	0.0286	0.0067	0.0074	0.0256	0.0122	1750 LN	0.0000	0.0023	0.0218	0.0168	0.0074	0.0897	0.0244
7500 AS	0.0054	0.0125	0.0150	0.0134	0.0074	0.0000	0.0000	0150 MA	0.0002	0.0015	0.0109	0.0268	0.0741	0.0897	0.1951
6180 AV	0.0249	0.0224	0.0367	0.0503	0.0222	0.0385	0.0122	4700 ML	0.0007	0.0008	0.0027	0.0000	0.0074	0.0000	0.0000
6400 AW	0.0093	0.0015	0.0054	0.0101	0.0000	0.0128	0.0244	3700 NM	0.0335	0.0379	0.0395	0.0201	0.0370	0.0128	0.0244
7400 AZ	0.0121	0.0159	0.0109	0.0101	0.0296	0.0256	0.0000	0900 NN	0.0005	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0593	0.1578	0.1578	0.1141	0.0667	0.0256	0.0244	3900 NR	0.0114	0.0156	0.0109	0.0134	0.0074	0.0128	0.0122
4000 BT	0.0171	0.0231	0.0259	0.0134	0.0000	0.0128	0.0000	2200 MS	0.0258	0.0254	0.0340	0.0168	0.0148	0.0000	0.0000
6000 CN	0.0008	0.0008	0.0000	0.0034	0.0000	0.0000	0.0000	3300 NU	0.0029	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0034	0.0008	0.0000	0.0034	0.0000	0.0128	0.0000	1400 NC	0.0001	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTI	0.0018	0.0027	0.0014	0.0101	0.0074	0.0128	0.0122	0300 OS	0.0244	0.0064	0.0054	0.0067	0.0074	0.0769	0.0366
1633 CTM	0.0007	0.0011	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0023	0.0027	0.0054	0.0034	0.0222	0.0000	0.0122
1644 CTO	0.0024	0.0004	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0054	0.0114	0.0027	0.0067	0.0000	0.0000	0.0244
1655 CTR	0.0015	0.0015	0.0014	0.0034	0.0000	0.0000	0.0122	7600 PH	0.0021	0.0027	0.0000	0.0034	0.0000	0.0000	0.0000
1611 CTT	0.0148	0.0023	0.0027	0.0067	0.0074	0.0000	0.0000	1080 PI	0.0033	0.0030	0.0136	0.0168	0.0074	0.0128	0.0122
5080 CU	0.0064	0.0023	0.0000	0.0067	0.0074	0.0128	0.0000	4600 PM	0.0005	0.0004	0.0041	0.0000	0.0074	0.0000	0.0122
8300 DN	0.0078	0.0102	0.0068	0.0134	0.0074	0.0128	0.0000	1800 PN	0.0226	0.0224	0.0204	0.0168	0.0148	0.0256	0.0122
2100 DK	0.0068	0.0083	0.0150	0.0000	0.0074	0.0000	0.0000	7000 PR	0.0032	0.0008	0.0054	0.0067	0.0148	0.0385	0.0244
3200 DM	0.0006	0.0023	0.0054	0.0034	0.0074	0.0000	0.0000	0200 QM	0.0187	0.0220	0.0245	0.0201	0.0148	0.0256	0.0122
1900 DP	0.0062	0.0064	0.0095	0.0034	0.0074	0.0128	0.0122	1500 RM	0.0213	0.0288	0.0163	0.0134	0.0074	0.0256	0.0000
1010 DS	0.0013	0.0011	0.0041	0.0101	0.0148	0.0000	0.0122	2500 RP	0.0020	0.0004	0.0000	0.0000	0.0074	0.0000	0.0122
4100 EM	0.0371	0.0835	0.0558	0.0872	0.0667	0.0000	0.0488	3600 SN	0.0814	0.0258	0.0245	0.0268	0.0370	0.0513	0.0122
3800 EN	0.0428	0.0398	0.0245	0.0201	0.0370	0.0513	0.0122	2400 SH	0.0076	0.0008	0.0000	0.0101	0.0000	0.0000	0.0000
5380 EQ	0.0063	0.0008	0.0027	0.0000	0.0074	0.0000	0.0000	2000 SK	0.0191	0.0315	0.0327	0.0403	0.0222	0.0000	0.0000
1000 ET	0.0080	0.0102	0.0177	0.0369	0.0296	0.0385	0.0610	0250 SM	0.0087	0.0080	0.0095	0.0034	0.0074	0.0000	0.0000
0350 EW	0.0023	0.0011	0.0041	0.0000	0.0000	0.0128	0.0122	0400 ST	0.0087	0.0053	0.0082	0.0101	0.0370	0.0128	0.0366
5000 FN	0.0707	0.0159	0.0109	0.0067	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0124	0.0190	0.0354	0.0201	0.0148	0.0513	0.0488	0500 TM	0.0070	0.0091	0.0041	0.0067	0.0000	0.0000	0.0000
0600 GM	0.0205	0.0319	0.0218	0.0537	0.0444	0.0256	0.0488	5800 UT	0.0043	0.0011	0.0014	0.0000	0.0222	0.0000	0.0000
								1700 YN	0.0334	0.0455	0.0218	0.0302	0.0222	0.0000	0.0244

TABLE B-4 (Continued)

## ACTIVE MARINER PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0389	0.0039	0.0104	0.0000	0.0256	0.0000	0.0000	4400 GS	0.0008	0.0026	0.0035	0.0000	0.0000	0.0000	0.0000
6700 AB	0.0537	0.0636	0.0519	0.0182	0.0513	0.0227	0.0000	8000 HM	0.0270	0.0233	0.0000	0.0000	0.0000	0.0000	0.0000
5600 AC	0.0016	0.0026	0.0035	0.0182	0.0000	0.0000	0.0000	4300 HT	0.0586	0.0324	0.0208	0.0182	0.0256	0.0000	0.0000
6080 AF	0.0610	0.0571	0.0519	0.0364	0.1026	0.0000	0.0000	2300 IS	0.0012	0.0039	0.0000	0.0000	0.0000	0.0227	0.0000
7100 AG	0.0033	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2600 JO	0.0004	0.0026	0.0000	0.0000	0.0000	0.0000	0.0000
7300 AK	0.0139	0.0246	0.0104	0.0364	0.0000	0.0000	0.0000	3100 LI	0.0037	0.0065	0.0035	0.0000	0.0000	0.0227	0.0000
6500 AO	0.0193	0.0233	0.0242	0.0182	0.0256	0.0000	0.0000	1750 LN	0.0000	0.0000	0.0173	0.0000	0.0000	0.0455	0.0385
7500 AS	0.0053	0.0130	0.0138	0.0000	0.0256	0.0227	0.0000	0150 MA	0.0000	0.0013	0.0000	0.0727	0.0256	0.1136	0.1923
6180 AV	0.0258	0.0311	0.0484	0.0364	0.0256	0.0000	0.0000	4700 ML	0.0008	0.0039	0.0000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.0053	0.0091	0.0069	0.0182	0.0256	0.0682	0.0000	3700 MN	0.0373	0.0376	0.0415	0.0727	0.0000	0.0909	0.1923
7400 AZ	0.0168	0.0208	0.0208	0.0000	0.0256	0.0000	0.0385	0900 NN	0.0008	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0889	0.1595	0.1903	0.1818	0.1282	0.0909	0.0385	3900 MR	0.0143	0.0195	0.0173	0.0182	0.0256	0.0000	0.0385
4000 BT	0.0164	0.0376	0.0277	0.0000	0.0000	0.0227	0.0000	2200 MS	0.0266	0.0246	0.0242	0.0182	0.0000	0.0455	0.0000
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0004	0.0000	0.0000	0.0182	0.0000	0.0000	0.0000	1400 NC	0.0004	0.0000	0.0000	0.0000	0.0256	0.0455	0.0769
1666 CTI	0.0008	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0300 OS	0.0352	0.0039	0.0069	0.0182	0.0000	0.0000	0.0000
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0000	0.0026	0.0069	0.0000	0.0000	0.0000	0.0000
1644 CTO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0029	0.0026	0.0035	0.0182	0.0256	0.0000	0.0000
1655 CTR	0.0000	0.0013	0.0000	0.0182	0.0000	0.0000	0.0000	7600 PH	0.0004	0.0026	0.0000	0.0000	0.0256	0.0227	0.0000
1611 CTT	0.0074	0.0013	0.0000	0.0000	0.0000	0.0227	0.0000	1080 PI	0.0012	0.0052	0.0000	0.0000	0.0256	0.0227	0.0000
5080 CU	0.0008	0.0013	0.0000	0.0000	0.0000	0.0455	0.0000	4600 PM	0.0004	0.0000	0.0000	0.0182	0.0000	0.0000	0.0000
8300 DN	0.0004	0.0065	0.0035	0.0000	0.0000	0.0000	0.0000	1800 PN	0.0152	0.0130	0.0173	0.0000	0.0256	0.0227	0.0000
2100 DK	0.0082	0.0143	0.0104	0.0000	0.0256	0.0227	0.0000	7000 PR	0.0012	0.0026	0.0035	0.0000	0.0256	0.0227	0.0385
3200 DM	0.0000	0.0052	0.0035	0.0000	0.0000	0.0000	0.0000	0200 QM	0.0209	0.0156	0.0069	0.0182	0.0256	0.0000	0.0000
1900 DP	0.0025	0.0013	0.0069	0.0000	0.0000	0.0000	0.0000	1500 RM	0.0147	0.0143	0.0138	0.0000	0.0000	0.0000	0.0000
1010 DS	0.0000	0.0052	0.0000	0.0182	0.0000	0.0227	0.0000	2500 RP	0.0025	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.0242	0.0700	0.0830	0.0545	0.0256	0.0455	0.0385	3600 SN	0.1012	0.0169	0.0069	0.0000	0.0513	0.0000	0.0000
3800 EN	0.0463	0.0545	0.0623	0.0182	0.0256	0.0227	0.0000	2490 SH	0.0107	0.0000	0.0000	0.0182	0.0000	0.0000	0.0000
5380 EQ	0.0016	0.0013	0.0000	0.0000	0.0000	0.0000	0.0000	2000 SK	0.0168	0.0208	0.0346	0.0000	0.0513	0.0000	0.0000
1000 ET	0.0016	0.0065	0.0277	0.0909	0.0256	0.0455	0.0385	0250 SM	0.0156	0.0208	0.0346	0.0000	0.0513	0.0000	0.0000
0350 EW	0.0020	0.0026	0.0000	0.0000	0.0000	0.0000	0.0385	0400 ST	0.0041	0.0052	0.0138	0.0364	0.0256	0.0000	0.0000
5000 FN	0.0733	0.0117	0.0000	0.0000	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0098	0.0195	0.0346	0.0000	0.0000	0.0455	0.1154	0500 TM	0.0037	0.0039	0.0000	0.0000	0.0256	0.0000	0.0000
0600 GM	0.0307	0.0441	0.0242	0.0364	0.0513	0.0000	0.0385	5800 UT	0.0012	0.0013	0.0000	0.0182	0.0000	0.0000	0.0000
								1700 YN	0.0197	0.0298	0.0311	0.0364	0.0256	0.0455	0.0769

TABLE B-4 (Continued)

## 526YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0751	0.0137	0.0228	0.0242	0.0192	0.0294	0.0000	4400 GS	0.0039	0.0017	0.0000	0.0161	0.0000	0.0000	0.0000
6700 AB	0.0105	0.0274	0.0274	0.0000	0.0000	0.0000	0.0000	8030 HM	0.0353	0.0120	0.0183	0.0081	0.0000	0.0000	0.0000
5600 AC	0.0033	0.0017	0.0046	0.0000	0.0192	0.0000	0.0000	4300 HT	0.0221	0.0172	0.0137	0.0161	0.0000	0.0294	0.0000
6000 AF	0.0143	0.0172	0.0183	0.0081	0.0192	0.0000	0.0000	2300 IS	0.0017	0.0034	0.0046	0.0081	0.0000	0.0294	0.0476
7100 AG	0.0017	0.0051	0.0000	0.0000	0.0000	0.0000	0.0000	2600 JO	0.0006	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000
7300 AK	0.0055	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3100 LI	0.0011	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000
5500 AO	0.0061	0.0086	0.0000	0.0161	0.0385	0.0000	0.0000	1750 LN	0.0000	0.0017	0.0091	0.0081	0.0192	0.0588	0.0000
7500 AS	0.0017	0.0034	0.0046	0.0000	0.0000	0.0000	0.0000	0150 MA	0.0000	0.0034	0.0091	0.0242	0.0385	0.0000	0.0000
6100 AV	0.0143	0.0086	0.0000	0.0081	0.0192	0.0294	0.0000	4700 ML	0.0000	0.0000	0.0274	0.0242	0.0000	0.0000	0.0476
6400 AW	0.0017	0.0051	0.0137	0.0242	0.0577	0.0000	0.0000	3700 NM	0.0044	0.0172	0.0000	0.0242	0.0000	0.0000	0.0000
4000 AZ	0.0061	0.0034	0.0046	0.0081	0.0385	0.0000	0.0000	0900 NN	0.0006	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0574	0.1784	0.1096	0.0565	0.0192	0.0294	0.0476	3900 NR	0.0094	0.0069	0.0046	0.0645	0.0000	0.0294	0.0000
4000 BT	0.0039	0.0086	0.0137	0.0081	0.0000	0.0000	0.0476	2200 MS	0.0094	0.0189	0.0091	0.0242	0.0192	0.0000	0.0000
6000 CN	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0006	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0081	0.0000	0.0882	0.0476
1666 CTI	0.0006	0.0017	0.0000	0.0000	0.0000	0.0000	0.0300	0300 OS	0.0121	0.0103	0.0091	0.0000	0.0000	0.0000	0.0000
1633 CTM	0.0000	0.0017	0.0046	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0022	0.0103	0.0000	0.0081	0.0000	0.0000	0.0000
1644 CTO	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0022	0.0031	0.0000	0.0000	0.0000	0.0000	0.0000
1655 CTR	0.0028	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7500 PH	0.0006	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000
1611 CTT	0.0050	0.0034	0.0000	0.0000	0.0000	0.0000	0.0000	1080 PI	0.0022	0.0017	0.0091	0.0000	0.0192	0.0000	0.0000
5000 CU	0.0039	0.0017	0.0000	0.0000	0.0000	0.0294	0.0000	4500 PM	0.0011	0.0000	0.0000	0.0081	0.0000	0.0294	0.0000
8300 DN	0.0039	0.0051	0.0000	0.0000	0.0000	0.0000	0.0000	1800 PN	0.0132	0.0154	0.0137	0.0000	0.0000	0.0000	0.0000
2100 DK	0.0022	0.0086	0.0320	0.0081	0.0000	0.0000	0.0000	7000 PR	0.0006	0.0034	0.0000	0.0081	0.0192	0.0000	0.0000
3200 DM	0.0011	0.0051	0.0046	0.0000	0.0000	0.0000	0.0000	0200 QM	0.0110	0.0350	0.0228	0.0403	0.0192	0.0000	0.0000
1900 DP	0.0022	0.0069	0.0046	0.0000	0.0000	0.0000	0.0000	1500 RM	0.0243	0.0172	0.0000	0.0081	0.0000	0.0000	0.0000
1010 DS	0.0028	0.0069	0.0091	0.0081	0.0000	0.0000	0.0000	2500 RP	0.0011	0.0017	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.0243	0.1029	0.0502	0.1048	0.0385	0.0000	0.0476	3500 SN	0.3013	0.0446	0.0000	0.0161	0.0769	0.0000	0.0476
3800 EN	0.0021	0.0240	0.0228	0.0161	0.0192	0.0000	0.0476	2490 SH	0.0000	0.0000	0.0046	0.0000	0.0000	0.0000	0.0000
5300 EQ	0.0022	0.0017	0.0046	0.0000	0.0000	0.0000	0.0000	2000 SK	0.0088	0.0240	0.0137	0.0081	0.0000	0.0000	0.0000
1000 ET	0.0127	0.0360	0.1005	0.1774	0.0769	0.0852	0.0952	0250 SM	0.0066	0.0189	0.0137	0.0161	0.0385	0.0000	0.0000
0350 EW	0.0039	0.0120	0.0046	0.0000	0.0000	0.0000	0.0476	0400 ST	0.0408	0.0257	0.0365	0.0161	0.0385	0.0000	0.1905
5000 FN	0.0000	0.0309	0.0137	0.0081	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0298	0.0892	0.1872	0.1210	0.2308	0.2941	0.2857	0500 TM	0.0072	0.0069	0.0091	0.0000	0.0000	0.0000	0.0000
0600 GM	0.0469	0.0395	0.0137	0.0565	0.0962	0.0294	0.0000	5800 UT	0.0028	0.0000	0.0046	0.0081	0.0000	0.0000	0.0000
								1700 YN	0.0226	0.0309	0.0228	0.0000	0.0000	0.0000	0.0000

TABLE B-4 (Continued)

## PRIOR SERVICE PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0004	0.0000	0.0030	0.0122	0.0164	0.0000	0.0000	4400 GS	0.0125	0.0000	0.0284	0.0517	0.0164	0.0333	0.0100
6700 AB	0.0121	0.0000	0.0436	0.0122	0.0164	0.0167	0.0200	8000 HM	0.0462	0.0357	0.0355	0.0304	0.0164	0.0167	0.0200
6600 AC	0.0125	0.0000	0.0000	0.0030	0.0000	0.0000	0.0000	4300 HT	0.0575	0.0357	0.0213	0.0182	0.0082	0.0417	0.0000
6000 AF	0.0156	0.0357	0.0426	0.0334	0.0410	0.0167	0.0200	2300 IS	0.0026	0.0000	0.0071	0.0030	0.0082	0.0083	0.0100
7100 AG	0.0009	0.0000	0.0000	0.0030	0.0000	0.0000	0.0100	2600 JO	0.0022	0.0000	0.0000	0.0000	0.0082	0.0000	0.0000
7300 AK	0.0069	0.0357	0.0000	0.0091	0.0164	0.0083	0.0100	3100 LI	0.0056	0.0000	0.0000	0.0030	0.0000	0.0083	0.0000
6500 AO	0.0376	0.0000	0.0071	0.0182	0.0164	0.0333	0.0000	1750 LN	0.0013	0.0357	0.0000	0.0030	0.0246	0.0167	0.0100
7500 AS	0.0086	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100	0150 MA	0.0086	0.0000	0.0000	0.0061	0.0328	0.1250	0.1500
6100 AV	0.0441	0.0000	0.0567	0.0152	0.0246	0.0083	0.0400	4700 ML	0.0013	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100
6400 AW	0.0151	0.0000	0.0284	0.0426	0.0246	0.0250	0.0000	3700 NM	0.0320	0.0357	0.0142	0.0304	0.0328	0.0250	0.0500
7400 AZ	0.0168	0.0000	0.0142	0.0061	0.0000	0.0167	0.0200	0900 NN	0.0043	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0367	0.1071	0.0213	0.0243	0.0333	0.0417	0.0000	3900 NR	0.0143	0.0000	0.0000	0.0152	0.0246	0.0167	0.0100
4000 BT	0.0229	0.0000	0.0567	0.0152	0.0000	0.0000	0.0100	2200 MS	0.0173	0.0000	0.0993	0.0213	0.0082	0.0083	0.0100
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100	3300 MU	0.0030	0.0000	0.0071	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0030	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100	1400 NC	0.0017	0.0000	0.0000	0.0000	0.0000	0.0083	0.0300
1665 CTT	0.0039	0.0000	0.0071	0.0030	0.0000	0.0000	0.0100	0300 OS	0.0475	0.0000	0.0355	0.0334	0.0164	0.0083	0.0000
1633 CTM	0.0039	0.0357	0.0000	0.0213	0.0000	0.0000	0.0100	0450 OT	0.0112	0.0357	0.0355	0.0182	0.0410	0.0417	0.0300
1644 CTO	0.0022	0.0000	0.0000	0.0061	0.0000	0.0000	0.0000	2700 PC	0.0043	0.0000	0.0000	0.0030	0.0000	0.0000	0.0000
1655 CTR	0.0095	0.0000	0.0000	0.0030	0.0164	0.0000	0.0000	7600 PH	0.0004	0.0000	0.0000	0.0030	0.0082	0.0000	0.0000
1611 CTT	0.0052	0.0000	0.0000	0.0030	0.0082	0.0083	0.0000	1080 PI	0.0073	0.0000	0.0000	0.0000	0.0246	0.0000	0.0100
5080 CU	0.0117	0.0000	0.0000	0.0122	0.0164	0.0250	0.0000	4600 PM	0.0009	0.0000	0.0000	0.0000	0.0082	0.0083	0.0000
8300 DN	0.0078	0.0357	0.0213	0.0122	0.0164	0.0083	0.0000	1800 PN	0.0043	0.0357	0.0071	0.0152	0.0082	0.0250	0.0100
2100 DK	0.0086	0.0000	0.0000	0.0061	0.0082	0.0000	0.0000	7000 PR	0.0065	0.0357	0.0142	0.0152	0.0410	0.0167	0.0500
3200 DM	0.0017	0.0000	0.0000	0.0000	0.0000	0.0083	0.0000	0200 QM	0.0212	0.0357	0.0284	0.0182	0.0000	0.0250	0.0300
1900 DP	0.0121	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1500 RM	0.0587	0.0000	0.0142	0.0122	0.0000	0.0167	0.0000
1010 DS	0.0022	0.0357	0.0000	0.0030	0.0000	0.0167	0.0000	2500 RP	0.0069	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.0501	0.0357	0.0355	0.0456	0.0738	0.0083	0.0400	3600 SN	0.0017	0.1071	0.0284	0.0213	0.0164	0.0083	0.0400
3800 EN	0.0371	0.0000	0.0213	0.0365	0.0246	0.0250	0.0100	2490 SH	0.0056	0.0000	0.0000	0.0000	0.0000	0.0167	0.0000
5380 EQ	0.0099	0.0000	0.0071	0.0030	0.0082	0.0000	0.0100	2000 SK	0.0393	0.0357	0.0213	0.0182	0.0164	0.0333	0.0100
1000 ET	0.0281	0.0714	0.0355	0.0365	0.0410	0.0500	0.0600	0250 SM	0.0121	0.0000	0.0142	0.0061	0.0082	0.0000	0.0000
0350 EW	0.0035	0.0357	0.0000	0.0061	0.0164	0.0167	0.0100	0400 ST	0.0307	0.0714	0.0213	0.0608	0.0328	0.0333	0.0300
5000 FN	0.0000	0.0000	0.0071	0.0030	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0302	0.0357	0.0780	0.0669	0.0410	0.0250	0.1000	0500 TM	0.0048	0.0000	0.0000	0.0000	0.0000	0.0000	0.0200
0600 GM	0.0514	0.0000	0.0709	0.1155	0.1066	0.0667	0.0600	5800 UT	0.0104	0.0000	0.0071	0.0030	0.0164	0.0083	0.0000
								1700 YN	0.0043	0.0357	0.0000	0.0091	0.0082	0.0250	0.0100

TABLE B-4 (Continued)

## TAR PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.2857	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4400 GS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6700 AB	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	8000 HM	0.0000	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000
6600 AC	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4300 HT	0.0000	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000
6080 AF	0.0769	0.2000	0.0000	0.0000	0.2000	0.0000	0.0000	2300 IS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7100 AG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2600 JO	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7300 AK	0.0000	0.0000	0.1429	0.0000	0.0000	0.0000	0.0000	3100 LI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6500 AO	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1750 LN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7500 AS	0.0000	0.2667	0.0000	0.0000	0.0000	0.0000	0.0000	0150 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6180 AV	0.0220	0.2000	0.4286	0.2000	0.0000	0.0000	0.0000	4700 ML	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3700 NM	0.0110	0.0000	0.1429	0.0000	0.0000	0.0000	0.0000
7400 AZ	0.0220	0.0000	0.1429	0.0000	0.0000	0.0000	0.0000	0900 NN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3900 NR	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4000 BT	0.0000	0.0000	0.0000	0.0000	0.1000	0.0000	0.0000	2200 MS	0.0110	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	0.0000	0.1000	0.0000	0.0000
1622 CTA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0300 OS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0000	0.0002	0.0000	0.0000	0.0000	0.0000	0.0000
1644 CTO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1655 CTR	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7600 PH	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1611 CTT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1080 PI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5080 CU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.0000	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000	1800 PN	0.0110	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2100 DK	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7000 PR	0.0000	0.0000	0.1429	0.0000	0.0000	0.0000	0.0000
3200 DM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0200 QM	0.0000	0.0000	0.0000	0.0000	0.1000	0.0000	0.0000
1900 DP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1500 RM	0.0000	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000
1010 DS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2500 RP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.0330	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3600 SN	0.0000	0.0000	0.0000	0.2000	0.0000	0.0000	0.0000
3800 EN	0.0220	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2490 SH	0.2088	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5380 EQ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2000 SK	0.0000	0.0000	0.0000	0.2000	0.1000	0.0000	0.0000
1000 ET	0.0220	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000	0250 SM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0350 EW	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0400 ST	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5000 FN	0.1978	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0500 TM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0600 GM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	5800 UT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
								1700 YN	0.0220	0.0000	0.0000	0.0000	0.2000	0.2000	0.0000

TABLE B-4 (Continued)

LOS &gt; 9

	7800 AN	6700 AB	5600 AC	5080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
9-10	0.0122	0.0244	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0366	0.0122	0.0000	0.0366	0.0000	0.0000
10-11	0.0000	0.0074	0.0000	0.0074	0.0074	0.0147	0.0147	0.0074	0.0368	0.0147	0.0147	0.0147	0.0074	0.0000
11-12	0.0000	0.0109	0.0000	0.0217	0.0000	0.0109	0.0000	0.0109	0.0435	0.0109	0.0326	0.0217	0.0109	0.0000
12-13	0.0000	0.0133	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0267	0.0000	0.0133	0.0267	0.0000	0.0000
13-14	0.0000	0.0000	0.0000	0.0526	0.0000	0.0000	0.0000	0.0000	0.0526	0.0132	0.0000	0.0526	0.0000	0.0000
14-15	0.0000	0.0145	0.0000	0.0000	0.0000	0.0000	0.0000	0.0145	0.0725	0.0000	0.0000	0.0290	0.0000	0.0000
15-16	0.0000	0.0000	0.0000	0.0179	0.0000	0.0000	0.0000	0.0179	0.1429	0.0000	0.0357	0.0179	0.0000	0.0000
16-17	0.0000	0.0000	0.0000	0.0256	0.0000	0.0000	0.0000	0.0000	0.1795	0.0000	0.0513	0.1026	0.0000	0.0000
17-18	0.0000	0.0000	0.0000	0.0426	0.0000	0.0000	0.0000	0.0000	0.1489	0.0000	0.0213	0.1064	0.0000	0.0000
18-19	0.0000	0.0000	0.0000	0.0400	0.0000	0.0000	0.0000	0.0400	0.1600	0.0000	0.0800	0.0400	0.0000	0.0000
19-20	0.0833	0.0000	0.0000	0.0833	0.0000	0.1667	0.0000	0.0000	0.0833	0.0000	0.0833	0.0000	0.0000	0.0000
20-21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.0000	0.0000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000
21-22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22-23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
23-24	0.0000	0.0000	0.0000	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24-25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
25-26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
26-27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27-28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28-29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30-31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-4 (Continued)

LOS &gt; 9

	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
9-10	0.0000	0.0000	0.0000	0.0000	0.0122	0.0122	0.0000	0.0000	0.0122	0.0000	0.0244	0.0244	0.0732	0.0122
10-11	0.0000	0.0074	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0221	0.0074
11-12	0.0109	0.0000	0.0000	0.0000	0.0000	0.0000	0.0109	0.0000	0.0000	0.0000	0.0435	0.0217	0.0326	0.0109
12-13	0.0000	0.0133	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0133	0.0000	0.0133	0.0133	0.0400	0.0133
13-14	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0132	0.0395	0.0000	0.0132	0.0000
14-15	0.0000	0.0000	0.0290	0.0000	0.0000	0.0000	0.0145	0.0000	0.0000	0.0000	0.0290	0.0000	0.0000	0.0435
15-16	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0179	0.0000	0.0000	0.0000	0.0000	0.0000	0.0179	0.0000
16-17	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.0000	0.0000	0.0000	0.0513	0.0000	0.0000	0.0256
17-18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0638	0.0000	0.0213	0.0000
18-19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0400	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
19-20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.0000
20-21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
21-22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22-23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
23-24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24-25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
25-26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
26-27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27-28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28-29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30-31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



TABLE B-4 (Continued)

LOS &gt; 9

	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
9-10	0.0000	0.0732	0.0122	0.0000	0.0610	0.0488	0.0122	0.0000	0.0122	0.0122	0.0000	0.0000	0.0122	0.0976
10-11	0.0000	0.0956	0.0000	0.0000	0.1029	0.0515	0.0074	0.0000	0.0074	0.0000	0.0000	0.0000	0.0147	0.1544
11-12	0.0000	0.0543	0.0000	0.0000	0.1413	0.0543	0.0000	0.0109	0.0000	0.0000	0.0000	0.0000	0.0000	0.1196
12-13	0.0000	0.0667	0.0133	0.0000	0.1333	0.0267	0.0000	0.0133	0.0000	0.0133	0.0000	0.0000	0.0133	0.1200
13-14	0.0000	0.0395	0.0132	0.0000	0.1711	0.0263	0.0000	0.0000	0.0000	0.0000	0.0132	0.0000	0.0000	0.1053
14-15	0.0000	0.0435	0.0000	0.0000	0.1014	0.0290	0.0000	0.0000	0.0145	0.0000	0.0000	0.0000	0.0000	0.1304
15-16	0.0179	0.0179	0.0357	0.0179	0.0536	0.0000	0.0000	0.0000	0.0000	0.0179	0.0000	0.0000	0.0000	0.0714
16-17	0.0000	0.0256	0.0256	0.0000	0.0513	0.0513	0.0000	0.0256	0.0000	0.0000	0.0000	0.0000	0.0000	0.0513
17-18	0.0000	0.0851	0.0213	0.0000	0.1064	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0638
18-19	0.0000	0.0000	0.0000	0.0000	0.0400	0.0400	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1200
19-20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
20-21	0.0000	0.0000	0.0000	0.0000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
21-22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22-23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
23-24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24-25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000
25-26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
26-27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27-28	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28-29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30-31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



TABLE B-4 (Continued)

LOS &gt; 9

	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
9-10	0.0000	0.0366	0.0000	0.0244	0.0122	0.0000	0.1585	0.0122	0.0000	0.0000	0.0000	0.0244	0.0000	0.0122
10-11	0.0000	0.0074	0.0074	0.0000	0.0074	0.0000	0.1029	0.0000	0.0000	0.0074	0.0000	0.0294	0.0074	0.0147
11-12	0.0000	0.0217	0.0000	0.0109	0.0326	0.0000	0.0870	0.0000	0.0000	0.0109	0.0000	0.0326	0.0109	0.0000
12-13	0.0000	0.0000	0.0133	0.0000	0.0133	0.0000	0.1867	0.0000	0.0133	0.0000	0.0000	0.0000	0.0000	0.0533
13-14	0.0000	0.0000	0.0000	0.0000	0.0132	0.0000	0.2237	0.0000	0.0132	0.0000	0.0000	0.0132	0.0000	0.0000
14-15	0.0000	0.0000	0.0000	0.0290	0.0290	0.0000	0.2174	0.0145	0.0145	0.0000	0.0000	0.0145	0.0145	0.0145
15-16	0.0000	0.0000	0.0000	0.0179	0.0179	0.0000	0.2321	0.0000	0.0179	0.0000	0.0000	0.0179	0.0000	0.0357
16-17	0.0000	0.0256	0.0256	0.0256	0.0256	0.0000	0.0256	0.0000	0.0513	0.0000	0.0000	0.0000	0.0000	0.0000
17-18	0.0000	0.0213	0.0000	0.0213	0.0426	0.0000	0.1702	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213
18-19	0.0000	0.0000	0.0000	0.0000	0.0400	0.0000	0.2400	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
19-20	0.0000	0.0833	0.0000	0.0000	0.0000	0.0000	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833
20-21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
21-22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250
22-23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
23-24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111
24-25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
25-26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667
26-27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27-28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28-29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
30-31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-4 (Continued)

LOS &gt; 9

	7000 PK	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN
9-10	0.0000	0.0366	0.0122	0.0000	0.0244	0.0000	0.0000	0.0000	0.0122	0.0000	0.0000	0.0000	0.0000
10-11	0.0368	0.0294	0.0294	0.0000	0.0221	0.0000	0.0074	0.0000	0.0441	0.0000	0.0147	0.0000	0.0147
11-12	0.0109	0.0000	0.0109	0.0000	0.0000	0.0109	0.0217	0.0000	0.0217	0.0000	0.0000	0.0109	0.0217
12-13	0.0000	0.0400	0.0133	0.0000	0.0133	0.0000	0.0133	0.0000	0.0267	0.0000	0.0000	0.0000	0.0267
13-14	0.0132	0.0000	0.0263	0.0000	0.0395	0.0000	0.0000	0.0000	0.0132	0.0000	0.0132	0.0132	0.0132
14-15	0.0000	0.0290	0.0000	0.0000	0.0000	0.0000	0.0145	0.0000	0.0000	0.0000	0.0145	0.0000	0.0290
15-16	0.0179	0.0000	0.0000	0.0000	0.0000	0.0357	0.0179	0.0000	0.0179	0.0000	0.0000	0.0536	0.0179
16-17	0.0256	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.0000	0.0000	0.0256	0.0513
17-18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213	0.0213	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
18-19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
19-20	0.0000	0.0000	0.0000	0.0000	0.0833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
20-21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
21-22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22-23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250
23-24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
24-25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
25-26	0.0000	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500
26-27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27-28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28-29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30-31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-5

FY 1986 LOSS RATES BY  
ENLISTMENT PROGRAM, RATING AND LOS

## 4YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.2941	0.1451	0.6472	0.4462	0.2381	0.2857	0.3333	4400 GS	0.0000	0.0833	0.2770	0.0656	0.4429	0.0185	0.0789
6700 AB	0.2627	0.0694	0.5626	0.2145	0.1634	0.1074	0.1348	8000 HM	0.2142	0.0640	0.4802	0.2181	0.1534	0.0953	0.1445
6600 AC	0.0000	0.0429	0.1698	0.4464	0.0588	0.0323	0.1197	4300 HT	0.0619	0.0619	0.6256	0.2629	0.0962	0.0407	0.1193
6000 AF	0.1905	0.0472	0.4712	0.1396	0.0957	0.1273	0.1184	2300 IS	0.1958	0.0323	0.3895	0.2453	0.1087	0.1538	0.1400
7100 AG	0.1688	0.0450	0.5182	0.1733	0.1212	0.1887	0.1463	2600 JO	0.0000	0.1053	0.0800	0.4000	0.1304	0.0667	0.1250
7300 AK	0.1733	0.0876	0.2929	0.1274	0.0902	0.1574	0.1458	3100 LI	0.0000	0.0800	0.3704	0.1622	0.1818	0.1176	0.0714
6500 AO	0.2036	0.0927	0.4738	0.2017	0.1518	0.0928	0.1170	1750 LN	0.0000	0.0000	0.2857	0.2609	0.1154	0.2667	0.1026
7500 AS	0.1933	0.0333	0.5229	0.1500	0.1500	0.1563	0.1250	0150 MA	0.0000	0.0000	0.0000	0.0909	0.0556	0.0417	0.0357
6180 AV	0.1678	0.0554	0.5444	0.1810	0.1148	0.1113	0.1805	4700 ML	0.1176	0.0000	0.3684	0.1818	0.0000	0.0000	0.0000
6400 AW	0.2643	0.0333	0.4850	0.2036	0.0619	0.0132	0.0333	3700 NM	0.2012	0.0757	0.5605	0.1778	0.0875	0.0650	0.1686
7400 AZ	0.2405	0.0556	0.3841	0.1282	0.1290	0.1613	0.0776	0900 NN	0.2059	0.0526	0.4000	0.1905	0.1905	0.0714	0.0435
0100 BM	0.0000	0.0530	0.5248	0.2332	0.1216	0.0709	0.1182	3900 NR	0.2348	0.0699	0.6833	0.1712	0.0917	0.0482	0.0545
4000 BT	0.3066	0.0668	0.5927	0.2464	0.1126	0.0525	0.1624	2200 MS	0.3055	0.1060	0.4600	0.1514	0.1036	0.0714	0.1422
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.2875	0.1837	0.1765	0.0769	0.0000
1622 CTA	0.1798	0.0411	0.2088	0.1159	0.0652	0.0213	0.0204	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTH	0.0357	0.0114	0.3286	0.2766	0.0769	0.0345	0.0571	0300 OS	0.2221	0.0815	0.5764	0.2709	0.0691	0.1022	0.1607
1633 CTM	0.0000	0.0000	0.0435	0.0909	0.4615	0.1250	0.0000	0450 OT	0.2079	0.0940	0.3234	0.2672	0.1885	0.1250	0.0612
1644 CTO	0.2408	0.0489	0.3830	0.1524	0.1047	0.0714	0.2031	2700 PC	0.2868	0.0336	0.5581	0.2817	0.1667	0.1579	0.1034
1655 CTR	0.1858	0.0359	0.2358	0.2763	0.1071	0.0845	0.0645	7600 PH	0.0000	0.0513	0.0000	0.4400	0.3667	0.2609	0.0870
1611 CTT	0.0769	0.0471	0.2324	0.1759	0.0385	0.0885	0.0968	1080 PI	0.2077	0.0726	0.4530	0.1644	0.0732	0.0000	0.0909
5900 CU	0.0000	0.0938	0.0909	0.5000	0.2105	0.0714	0.1250	4600 PM	0.0500	0.0000	0.6111	0.1667	0.0000	0.0000	0.0000
8300 DN	0.0000	0.0482	0.0500	0.5181	0.3158	0.0952	0.0556	1800 PN	0.2381	0.0662	0.3488	0.1617	0.0788	0.0927	0.1087
2100 DK	0.2093	0.0478	0.2884	0.1434	0.1143	0.0840	0.1053	7000 PR	0.1782	0.0820	0.2875	0.1193	0.1429	0.1839	0.0909
3200 DM	0.0000	0.1429	0.3810	0.2174	0.0000	0.0667	0.0909	0200 QM	0.2076	0.1146	0.5191	0.1348	0.0331	0.0882	0.1463
1900 DP	0.0000	0.0593	0.0774	0.3947	0.2680	0.0875	0.1667	1500 RM	0.2431	0.0777	0.4815	0.1897	0.1165	0.0605	0.1026
1010 DS	0.0000	0.0000	0.1111	0.0000	0.0769	0.1667	0.1818	2500 RP	0.2808	0.1200	0.4124	0.1538	0.1163	0.0250	0.0000
4100 EM	0.2216	0.0596	0.5953	0.1479	0.0702	0.0360	0.1126	3600 SN	0.3294	0.1383	0.6767	0.4419	0.5000	0.5000	0.8095
3800 EN	0.2247	0.0660	0.5741	0.1706	0.0734	0.0643	0.1640	2490 SH	0.2970	0.0803	0.5490	0.1921	0.1319	0.0851	0.0809
5380 EQ	0.0000	0.1071	0.0244	0.5385	0.2174	0.0588	0.0588	2000 SK	0.2139	0.0655	0.4054	0.1455	0.1696	0.1051	0.0833
1000 ET	0.0000	0.0748	0.1346	0.1000	0.2813	0.1667	0.0952	0250 SM	0.3529	0.1283	0.4571	0.2270	0.0833	0.0426	0.1912
0350 EW	0.1801	0.0667	0.5474	0.1188	0.2549	0.1667	0.1111	0400 ST	0.2178	0.0601	0.3350	0.0941	0.3209	0.0685	0.0746
5000 FN	0.3069	0.1441	0.7236	0.5479	0.4211	0.6667	0.5000	7200 TD	0.0000	0.6250	0.7344	0.4286	0.2941	0.3158	0.5185
0800 FT	0.0000	0.0315	0.3436	0.1371	0.2185	0.0806	0.1000	0500 TM	0.2427	0.0550	0.5189	0.1958	0.0859	0.0567	0.1206
0600 GM	0.2756	0.0511	0.4874	0.1586	0.0474	0.0777	0.1313	5800 UT	0.0000	0.0833	0.0968	0.3871	0.1429	0.0833	0.1538
								1700 YN	0.2106	0.0618	0.3287	0.1697	0.1042	0.1125	0.1940

TABLE B-5 (Continued)

## ACTIVE MARINER PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0889	0.6791	0.4717	0.3333	0.4286	0.0000	0.6667	4400 GS	0.0000	0.0000	0.4444	0.2000	0.2222	0.0000	0.1667
6700 AB	0.0000	0.6543	0.2562	0.1084	0.1765	0.1000	0.1163	8000 HM	0.2137	0.4890	0.3262	0.1744	0.1392	0.1520	0.1154
6600 AC	0.2917	0.4000	0.0833	0.2903	0.2222	0.1765	0.1429	4300 HT	0.1738	0.5733	0.2696	0.1200	0.0963	0.1129	0.1370
6080 AF	0.1644	0.4368	0.2273	0.0906	0.1102	0.1537	0.0776	2300 IS	0.0741	0.5435	0.1282	0.0909	0.2143	0.1364	0.0000
7100 AG	0.0612	0.4107	0.0556	0.2143	0.1154	0.0833	0.1667	2600 JO	0.1250	0.5714	0.0000	0.5000	0.0000	0.0833	0.1429
7300 AK	0.1250	0.3582	0.1364	0.1644	0.0317	0.1452	0.0600	3100 LI	0.0000	0.0000	0.6667	0.2500	0.2500	0.0000	1.0000
6500 AO	0.2139	0.5275	0.2857	0.1797	0.0541	0.0986	0.1452	1750 LN	0.0000	0.0000	0.0000	0.5000	0.2000	0.3333	0.0000
7500 AS	0.1190	0.4375	0.1705	0.1500	0.0811	0.1892	0.1212	0150 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667
6180 AV	0.1675	0.4751	0.3029	0.0933	0.1306	0.1745	0.1854	4700 ML	0.0000	0.5000	0.3333	0.0000	0.0000	0.0000	0.0000
6400 AW	0.3021	0.3671	0.2623	0.1515	0.0000	0.0244	0.1385	3700 MM	0.2000	0.5381	0.2761	0.0722	0.0656	0.1639	0.1209
7400 AZ	0.3333	0.3571	0.0968	0.0435	0.1290	0.0800	0.0357	0900 NN	0.3448	0.2500	0.2308	0.1667	0.3333	0.0000	0.0000
0100 BM	0.0000	0.6509	0.2706	0.1429	0.0833	0.0963	0.0714	3900 MR	0.1964	0.5714	0.2769	0.0851	0.0968	0.0625	0.0000
4000 BT	0.2512	0.5960	0.3805	0.0854	0.0750	0.1382	0.1887	2200 MS	0.3560	0.4950	0.2283	0.1143	0.1319	0.1053	0.1019
6000 CN	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
1622 CTA	0.1429	0.1333	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
1666 CTI	0.0000	0.1667	0.0000	0.0000	0.0000	0.2500	0.0000	0300 OS	0.1987	0.5939	0.3359	0.1698	0.1798	0.1538	0.0513
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.2857	0.4737	0.2308	0.1579	0.1667	0.0000	0.1667
1644 CTO	0.1731	0.3182	0.0000	0.0000	0.2857	0.2500	0.0000	2700 PC	0.0000	0.5250	0.2667	0.1667	0.0000	0.1111	0.1667
1655 CTR	0.1739	0.2222	0.1034	0.1818	0.0000	0.0000	0.0000	7600 PH	0.2500	0.3636	0.3750	0.3333	0.2500	0.1176	0.1000
1611 CTT	0.0000	0.3846	0.2500	0.0909	0.0588	0.1429	0.3636	1080 PI	0.1739	0.5161	0.0625	0.0000	0.2000	0.0000	0.0000
5080 CU	0.1852	0.5949	0.2162	0.0370	0.0000	0.1818	0.0909	4600 PM	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.2000	0.3514	0.1429	0.3333	0.4286	0.1250	0.0833	1800 PN	0.2000	0.4194	0.1461	0.1354	0.0685	0.1650	0.0690
2100 DK	0.3333	0.3265	0.1698	0.0732	0.0952	0.1364	0.0000	7000 PR	0.2162	0.2500	0.2791	0.0588	0.0588	0.2917	0.0000
3200 DM	0.0000	0.4286	0.0000	0.0000	0.0000	0.0000	0.0000	0200 QM	0.2178	0.5985	0.2400	0.1017	0.0303	0.0571	0.1364
1900 DP	0.1667	0.2927	0.1176	0.3548	0.1200	0.1579	0.0625	1500 RM	0.2167	0.4621	0.2375	0.0574	0.0808	0.1316	0.1806
1010 DS	0.3333	0.0714	0.6364	0.1176	0.0000	0.0588	0.0000	2500 RP	0.0000	0.4667	0.1429	0.0000	0.1667	0.2857	0.2000
4100 EM	0.1711	0.5259	0.2615	0.0915	0.0764	0.1468	0.1216	3600 SN	0.1243	0.6520	0.4696	0.4754	0.3548	0.6087	1.0000
3800 EN	0.3333	0.5418	0.2824	0.1145	0.0682	0.0811	0.0769	2400 SH	0.4444	0.6667	0.2813	0.2000	0.0571	0.1698	0.0476
5380 EO	0.2200	0.6883	0.2639	0.2105	0.0741	0.1364	0.0400	2000 SK	0.2794	0.5102	0.2171	0.1418	0.1228	0.1412	0.0676
1000 ET	0.2249	0.4940	0.3611	0.0851	0.1250	0.1795	0.0333	0250 SM	0.4176	0.4762	0.3043	0.0800	0.1111	0.0000	0.0870
0350 EW	0.1250	0.6716	0.2414	0.1111	0.2857	0.0000	0.0000	0400 ST	0.2772	0.4128	0.0959	0.1475	0.1714	0.0000	0.0000
5000 FN	0.0935	0.6784	0.4000	0.3333	0.3333	0.3333	0.5000	7200 TD	0.0000	1.0000	0.2500	0.6667	1.0000	1.0000	0.0000
0800 FT	0.1676	0.4733	0.2110	0.0526	0.2250	0.2174	0.0000	0500 TM	0.2937	0.4737	0.1429	0.1071	0.0667	0.1522	0.0357
0600 GM	0.2330	0.5104	0.1656	0.1619	0.0822	0.1447	0.1212	5800 UT	0.1667	0.7260	0.2821	0.1500	0.0588	0.3333	0.0000
								1700 YN	0.2059	0.3333	0.1308	0.1212	0.1429	0.1091	0.1203

TABLE B-5 (Continued)

## 526YO PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0000	0.1146	0.5763	0.6111	0.4545	1.0000	1.0000	4400 GS	0.2129	0.0542	0.2328	0.1045	0.4945	0.0769	0.1064
6700 AB	0.0000	0.1250	0.3462	0.2258	0.2750	0.2143	0.0000	8000 HM	0.1193	0.0415	0.1779	0.1471	0.4000	0.1445	0.1121
6600 AC	0.1777	0.0294	0.0262	0.5445	0.1587	0.0581	0.1111	4300 HT	0.1885	0.0833	0.2214	0.1064	0.3210	0.0556	0.0588
6080 AF	0.2500	0.1481	0.3014	0.1806	0.0976	0.0962	0.1379	2300 IS	0.0000	0.0000	0.3636	0.1429	0.0000	0.1250	0.0000
7100 AG	0.0000	0.0000	0.6000	0.1818	0.4000	0.2000	0.0000	2600 JO	0.3011	0.1000	0.0513	0.3243	0.2857	0.1304	0.3043
7300 AK	0.0000	0.0000	0.4444	0.2000	0.0606	0.0435	0.3333	3100 LI	0.0000	1.0000	0.5000	0.5000	0.0000	0.0000	1.0000
6500 AO	0.0000	0.1111	0.3000	0.0526	0.0000	0.2083	0.0000	1750 LN	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
7500 AS	0.0000	0.2308	0.1538	0.2500	0.0000	0.0000	0.3333	0150 MA	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
6180 AV	0.1390	0.0338	0.2382	0.0791	0.3054	0.1442	0.2222	4700 ML	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.0000	0.0000	0.3889	0.2857	0.1000	0.1429	0.1429	3700 NM	0.1230	0.0637	0.0917	0.1659	0.4316	0.1031	0.1876
7400 AZ	0.0000	0.0714	0.3214	0.2692	0.0714	0.5000	0.0000	0900 NN	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
0100 BM	0.0313	0.1143	0.5795	0.2118	0.2281	0.3214	0.1071	3500 NR	0.0000	0.0000	0.3571	0.1667	0.0000	0.0000	0.1667
4000 BT	0.2045	0.0907	0.2042	0.2440	0.4520	0.0965	0.0417	2200 MS	0.0000	0.1180	0.1578	0.2311	0.1585	0.0909	0.1429
6000 CN	0.0000	0.2000	1.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	0.2000	0.0000	0.0000	0.0000
1622 CTA	0.0000	0.1250	0.0769	0.1429	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTI	1.0000	0.0000	0.1111	0.3750	0.0882	0.0000	0.0000	0300 OS	0.0476	0.1096	0.1127	0.4523	0.1097	0.0488	0.0000
1633 CTM	0.1656	0.0243	0.0269	0.0195	0.3259	0.1186	0.0891	0450 OT	0.1765	0.0870	0.1935	0.3333	0.2353	0.0000	0.2857
1644 CTO	0.0000	0.0400	0.3333	0.1111	0.0909	0.0667	0.3333	2700 PC	0.0000	0.0909	0.4615	0.1667	0.0000	0.0000	1.0000
1655 CTR	0.0000	0.0492	0.0536	0.1702	0.0625	0.0000	0.2500	7600 PH	0.1806	0.0495	0.0583	0.5931	0.2151	0.1707	0.1455
1611 CTT	0.0000	0.0185	0.0674	0.1707	0.0625	0.1667	0.2500	1080 PI	0.0000	0.0526	0.5714	0.0000	0.2000	0.0000	0.5000
5080 CU	0.1976	0.0946	0.1075	0.5300	0.1466	0.1200	0.0730	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.2454	0.0746	0.0728	0.3884	0.2756	0.1290	0.0968	1800 PN	0.0714	0.0652	0.2679	0.1176	0.1176	0.0455	0.1538
2100 DK	0.0000	0.0000	0.2353	0.0000	0.2000	0.1250	0.0000	7000 PR	0.0000	0.0909	0.1111	0.1765	0.2500	0.0000	0.0000
3200 DM	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.1111	0200 QM	0.0000	0.0612	0.5854	0.1000	0.1500	0.0625	0.0909
1900 DP	0.1117	0.0673	0.0809	0.4563	0.2689	0.1748	0.1158	1500 RM	0.1930	0.0540	0.1787	0.1321	0.3939	0.0901	0.0814
1010 DS	0.1746	0.0456	0.0740	0.0511	0.4884	0.2479	0.1043	3600 SN	0.2174	0.1475	0.5932	0.4154	0.7500	0.4000	0.0000
4100 EM	0.0540	0.0499	0.1515	0.1348	0.4197	0.0742	0.1614	2490 SH	0.0000	0.0909	0.5625	0.2222	0.4000	0.7143	0.4000
3800 EN	0.0000	0.0294	0.4906	0.2093	0.1333	0.0323	0.0909	2900 SK	0.0000	0.0485	0.3396	0.1667	0.1892	0.0435	0.1250
5380 EQ	0.2209	0.0796	0.0939	0.4988	0.2030	0.1318	0.0202	0250 SM	0.0000	0.0962	0.2344	0.3333	0.1515	0.0000	0.0000
1000 ET	0.1270	0.0456	0.0573	0.0588	0.4250	0.1539	0.1288	0400 ST	0.0000	0.0608	0.0769	0.0564	0.4175	0.0856	0.0618
0350 EW	0.1195	0.0407	0.1955	0.0625	0.4667	0.1959	0.0789	7200 TD	0.0000	0.0000	1.0000	0.3333	0.0000	0.0000	0.5000
5000 FN	0.2034	0.1645	0.6197	0.4286	1.0000	1.0000	0.0000	0500 TM	0.0000	0.0000	0.1793	0.3889	0.0930	0.0400	0.0000
0800 FT	0.1271	0.0478	0.0992	0.0784	0.4068	0.0876	0.1128	5800 UT	0.1909	0.0769	0.0909	0.4892	0.1781	0.2424	0.0909
0600 GM	0.0000	0.0370	0.2049	0.3131	0.0816	0.0417	0.1333	1700 YN	0.0000	0.1667	0.3205	0.1594	0.0938	0.0952	0.3158

TABLE B-5 (Continued)

## PRIOR SERVICE PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.2319	0.4000	0.1951	0.2281	0.3750	0.5000	0.1667	4400 GS	0.0000	0.0000	0.0000	0.2222	0.0556	0.0588	0.0952
6700 AB	0.1111	0.0000	0.0800	0.1207	0.1081	0.1533	0.0904	8000 HT	0.1469	0.0000	0.0988	0.1156	0.1376	0.0881	0.1278
6600 AC	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.1754	4300 HT	0.1667	0.0000	0.1806	0.1310	0.1250	0.1227	0.1503
6080 AF	0.2059	0.2143	0.1477	0.1049	0.0985	0.0964	0.1035	2300 IS	0.0000	0.0000	0.0000	0.0000	0.1000	0.0435	0.0909
7100 AG	0.0000	0.0000	0.5000	0.0000	0.1250	0.0000	0.1500	2600 JO	1.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.0909
7300 AK	0.2143	0.0000	0.0000	0.2381	0.0333	0.0959	0.0811	3100 LI	0.1667	0.0000	0.0000	0.1667	0.1667	0.2353	0.0769
6500 AO	0.1647	0.1111	0.1304	0.1158	0.1007	0.0962	0.1098	1700 LN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7500 AS	0.0455	0.0000	0.0000	0.0909	0.0833	0.1081	0.0741	0156 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0417
6180 AV	0.1512	0.1176	0.2203	0.1481	0.1243	0.1279	0.1312	4700 ML	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.3333	0.1250	0.0000	0.0444	0.0571	0.1563	0.0580	3700 MM	0.1429	0.1538	0.2075	0.1085	0.1272	0.0941	0.1354
7400 AZ	0.0000	0.0000	0.1111	0.0952	0.1163	0.1200	0.1000	0900 MN	0.0000	0.0000	0.5000	0.0000	0.0000	0.2727	0.0909
0100 BA	0.2321	0.0000	0.3000	0.1318	0.1486	0.1140	0.1323	3900 NR	0.0000	0.0000	0.0769	0.0256	0.0847	0.0588	0.0755
4000 BT	0.2771	0.0000	0.2759	0.1099	0.1927	0.1774	0.2244	2200 NS	0.1628	0.0909	0.0943	0.0862	0.2000	0.1226	0.1270
6000 CN	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 NU	0.0000	0.0000	0.0000	0.1250	0.0000	0.1429	0.2308
1622 CTA	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.2222	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTI	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.1304	0300 OS	0.1368	0.1667	0.2558	0.1226	0.0977	0.1011	0.1268
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0450 OT	0.2000	0.0000	0.2000	0.0000	0.0476	0.1923	0.0625
1644 CTO	0.0000	0.0000	0.5000	0.0000	0.0769	0.2143	0.0909	2700 PC	0.1538	0.0000	0.0000	0.0000	0.1667	0.1892	0.1071
1655 CTR	0.2857	0.0000	0.0000	0.1429	0.1250	0.0370	0.0816	7600 PH	0.0000	0.0000	0.0000	1.0000	0.0000	0.2632	0.0769
1611 CTT	0.1429	0.0000	0.0000	0.0000	0.1176	0.1154	0.0465	1080 PI	0.1667	0.0000	0.2500	0.0000	0.0000	0.1364	0.0714
5080 CU	0.0800	0.1111	0.1538	0.1429	0.0278	0.1429	0.1111	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250
8300 DN	0.0000	0.0000	0.1667	0.1538	0.0435	0.1250	0.0278	1800 PN	0.0000	0.3333	0.1875	0.1500	0.1667	0.1270	0.1512
2100 DK	0.1538	0.3333	0.0000	0.0769	0.1212	0.1905	0.2195	7000 PR	0.0909	0.0000	0.2222	0.0417	0.0714	0.0526	0.1034
3200 DM	0.0000	0.0000	0.0000	0.5000	0.1429	0.0000	0.0000	0200 QM	0.2093	0.0000	0.2222	0.1410	0.0704	0.0556	0.1333
1900 DP	0.0000	0.0000	0.0000	0.0000	0.2121	0.0462	0.0877	1500 RM	0.1988	0.1429	0.3506	0.1127	0.1055	0.1119	0.1101
1010 DS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.0435	2500 RP	0.0000	0.0000	0.0000	0.0500	0.0833	0.1250	0.1875
4100 EM	0.0636	0.0000	0.1321	0.1081	0.0578	0.1036	0.1575	3600 SN	0.2078	0.2162	0.1939	0.1297	0.2909	0.2041	0.3704
3800 EN	0.1548	0.2222	0.2000	0.0972	0.1400	0.1064	0.1126	2490 SH	0.1842	0.6667	0.5333	0.1842	0.1563	0.1484	0.1582
5380 EQ	0.1818	0.2222	0.2500	0.2143	0.0769	0.0625	0.0556	0250 SK	0.1594	0.1667	0.1034	0.1129	0.1121	0.1079	0.0956
1000 ET	0.0213	0.0000	0.0500	0.0606	0.0952	0.0645	0.0677	0250 SM	0.2683	0.0000	0.0667	0.0833	0.1167	0.1356	0.0941
0350 EW	0.1000	0.0000	0.1667	0.0000	0.0000	0.0667	0.2000	0400 ST	0.1163	0.1667	0.1290	0.0204	0.0606	0.0147	0.1579
5000 FN	0.2374	0.3333	0.1304	0.1556	0.3750	0.0349	0.5000	7200 TD	0.0000	1.0000	0.5000	0.0000	0.0000	0.3333	0.6923
0800 FT	0.1087	0.2500	0.0476	0.1282	0.0714	0.0349	0.0862	0500 TM	0.1000	0.0000	0.1818	0.0000	0.0714	0.1224	0.1463
0600 GM	0.1630	0.5000	0.0638	0.0394	0.0465	0.0909	0.1415	5800 UT	0.0769	0.1429	0.2000	0.2222	0.0000	0.0833	0.0769
								1700 YN	0.2222	0.0000	0.1765	0.0943	0.1458	0.1963	0.1083

TABLE B-5 (Continued)

## TAR PROGRAM

Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9	Rate	1-3	3-4	4-5	5-6	6-7	7-8	8-9
7800 AN	0.0000	0.0588	0.2500	0.0000	0.6667	0.0000	0.0000	4400 GS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6700 AB	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	8000 HT	0.2239	0.0000	0.0000	0.1154	0.0000	0.0000	0.0714
6600 AC	0.2308	0.0000	0.4444	0.0000	0.0000	0.0000	0.0000	4300 HT	0.1522	0.0000	0.1667	0.0435	0.1250	0.0000	0.0000
6080 AF	0.1100	0.0500	0.4029	0.1000	0.0745	0.0476	0.1373	2300 IS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7100 AG	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2600 JO	0.0000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000
7300 AK	0.2078	0.0526	0.4375	0.0769	0.1000	0.0000	0.0000	3100 LI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6500 AO	0.2895	0.0000	0.2400	0.1429	0.1333	0.0000	0.0000	1750 LN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7500 AS	0.2381	0.0000	0.2500	0.3333	0.0000	0.0000	0.2000	0150 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6180 AV	0.1569	0.0291	0.3185	0.1489	0.1707	0.0714	0.0909	4700 ML	0.0000	0.0630	0.0000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.2333	0.0000	0.6364	0.0588	0.3000	0.0000	0.1667	3700 MM	0.3235	0.1000	0.4167	0.0000	0.0000	0.0000	0.0000
7400 AZ	0.1818	0.0833	0.2000	0.2727	0.0714	0.0000	0.3333	0900 MN	0.1818	0.0000	0.0000	0.0000	0.0000	0.1429	0.0000
0100 BM	0.0000	0.0000	0.3333	0.0588	0.0476	0.0000	0.0000	3900 MR	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000
4000 BT	0.2778	0.0556	0.2222	0.0000	0.0000	0.0000	0.3636	2200 MS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6000 CN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	3300 MU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTH	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0300 OS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1633 CTM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1644 CTO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2700 PC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1655 CTR	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7600 PH	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1611 CTT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1080 PI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
5080 CU	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1800 PN	0.3137	0.0000	0.1875	0.1290	0.0179	0.0870	0.385
2100 DK	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	7000 PR	0.3103	0.0833	0.5000	0.2727	0.0000	0.0000	0.2500
3200 DM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0200 QM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1900 DP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1500 RM	0.0000	0.0000	1.0000	1.0000	0.0000	0.0000	0.0000
1010 DS	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2500 RP	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.2571	0.0000	0.3333	0.1111	0.0667	0.0000	0.1667	3600 SN	0.0000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000
3800 EN	0.0000	0.2000	0.2857	0.0714	0.0000	0.0000	0.0000	2490 SH	0.0000	0.0000	1.0000	0.0000	1.0000	0.0000	0.0000
5380 EQ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2000 SK	0.2885	0.0000	0.1875	0.0882	0.0476	0.0645	0.0909
1000 ET	0.3000	0.0000	0.2222	0.3333	0.0000	0.1667	0.0000	0250 SM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0350 EW	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0400 ST	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
5000 FN	0.0000	0.5000	1.0000	0.0000	0.0000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0500 TM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0600 GM	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	5800 UT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
								1700 YN	0.2358	0.0000	0.1481	0.0588	0.0000	0.0000	0.0714

TABLE B-5 (Continued)

LOS &gt; 9

	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
9-10	1.0000	0.1000	0.0800	0.0587	0.0682	0.0510	0.0719	0.0741	0.1036	0.0843	0.0250	0.1078	0.1592	0.0000
10-11	0.0000	0.0691	0.1118	0.0633	0.0704	0.0596	0.0720	0.0465	0.1208	0.1205	0.0855	0.0829	0.0968	0.0000
11-12	1.0000	0.0221	0.0571	0.0431	0.1282	0.0556	0.0510	0.0323	0.0586	0.0786	0.0588	0.0451	0.0712	0.0000
12-13	0.3333	0.0596	0.0779	0.0390	0.0714	0.0366	0.0341	0.0597	0.0596	0.0693	0.0744	0.0746	0.0520	0.0000
13-14	0.0000	0.0412	0.0505	0.0238	0.1053	0.0305	0.0511	0.0000	0.0506	0.0583	0.0630	0.0463	0.0739	0.0000
14-15	1.0000	0.0455	0.0377	0.0288	0.0426	0.0213	0.0533	0.0370	0.0336	0.0345	0.0085	0.0317	0.0556	0.0000
15-16	0.0000	0.0324	0.0120	0.0280	0.0571	0.0286	0.0741	0.0423	0.0205	0.0241	0.0407	0.0497	0.0324	0.0000
16-17	0.0000	0.0194	0.0455	0.0128	0.0000	0.0341	0.0374	0.0000	0.0241	0.0159	0.0125	0.0541	0.0317	0.0000
17-18	0.0000	0.0326	0.0000	0.0113	0.0476	0.0000	0.0508	0.0135	0.0131	0.0000	0.0267	0.0074	0.0157	0.0000
18-19	0.0000	0.0370	0.0000	0.0110	0.0000	0.0000	0.0306	0.0122	0.0062	0.0000	0.0135	0.0174	0.0404	0.0000
19-20	0.0000	0.0732	0.0286	0.0470	0.0345	0.0354	0.0211	0.0893	0.0435	0.0645	0.0429	0.0629	0.0278	0.0000
20-21	0.0000	0.3421	0.4138	0.3571	0.4375	0.3457	0.4783	0.4500	0.3663	0.4545	0.3621	0.3757	0.3786	0.0000
21-22	0.0000	0.3529	0.3333	0.2450	0.3158	0.3409	0.1591	0.3200	0.2966	0.3636	0.2500	0.2275	0.2708	0.0000
22-23	0.0000	0.3529	0.1875	0.2486	0.0000	0.1923	0.4333	0.3500	0.3155	0.4800	0.3143	0.2275	0.2708	0.0000
23-24	0.0000	0.3030	0.3333	0.2192	0.3333	0.2609	0.2727	0.1250	0.2252	0.3500	0.1500	0.1887	0.2222	0.0000
24-25	0.0000	0.1087	0.1000	0.1554	0.3636	0.3810	0.1220	0.2857	0.2124	0.4211	0.0833	0.1809	0.2143	0.0000
25-26	0.0000	0.0769	0.0000	0.1579	0.1429	0.2667	0.0000	0.5000	0.1127	0.1000	0.0833	0.1809	0.2593	0.0000
26-27	0.0000	0.1579	0.1667	0.1085	0.1667	0.2500	0.3000	0.2000	0.2121	0.4286	0.4000	0.1692	0.2895	0.0000
27-28	0.0000	0.1765	0.3077	0.1918	0.3333	0.2857	0.0000	0.3333	0.2167	0.3750	0.3333	0.2105	0.1739	0.0000
28-29	0.0000	0.1667	0.2000	0.0727	0.1250	0.2000	0.0833	0.0000	0.1194	0.0000	0.0000	0.2083	0.1667	0.0000
29-30	0.0000	0.0000	0.2000	0.2500	0.0000	0.2500	0.5000	0.4000	0.2200	0.5000	0.3333	0.1176	0.3750	0.0000
30-31	0.0000	0.8333	0.5000	0.7619	0.6667	0.6667	0.7000	0.6667	0.6061	1.0000	0.4000	0.6471	0.3000	0.0000



TABLE B-5 (Continued)

LOS &gt; 9

	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
9-10	0.0400	0.0667	0.1111	0.0357	0.0702	0.0625	0.1163	0.0517	0.0656	0.1429	0.0800	0.0541	0.1024	0.0560
10-11	0.0926	0.0645	0.1655	0.0435	0.0267	0.0180	0.0820	0.0826	0.0513	0.0870	0.0854	0.1486	0.1015	0.0870
11-12	0.0741	0.0794	0.0661	0.0502	0.0115	0.0000	0.0423	0.1158	0.0230	0.0714	0.0661	0.0843	0.0544	0.0385
12-13	0.0000	0.0811	0.0943	0.0417	0.0600	0.0580	0.0440	0.0556	0.0093	0.0000	0.0840	0.0719	0.0816	0.0391
13-14	0.0000	0.1429	0.0656	0.0308	0.0000	0.0294	0.0130	0.0408	0.0182	0.0000	0.0891	0.0756	0.0587	0.0467
14-15	0.0000	0.0000	0.0385	0.0455	0.0667	0.0625	0.0423	0.0426	0.0357	0.0833	0.0532	0.0099	0.0624	0.0402
15-16	0.0333	0.0435	0.0000	0.0213	0.0426	0.1020	0.0189	0.0154	0.0244	0.0000	0.0145	0.0345	0.0285	0.0313
16-17	0.0000	0.0370	0.0385	0.0012	0.0638	0.0238	0.0095	0.0149	0.0000	0.0000	0.0600	0.0000	0.0244	0.0246
17-18	0.0625	0.0000	0.0417	0.0465	0.0600	0.0000	0.0000	0.0000	0.0000	0.0000	0.0172	0.0000	0.0073	0.0135
18-19	0.0000	0.0000	0.0204	0.0376	0.0000	0.0750	0.0000	0.0172	0.0111	0.0000	0.0577	0.0250	0.0070	0.0067
19-20	0.0000	0.0000	0.0377	0.1538	0.1053	0.1081	0.0781	0.0526	0.0337	0.3333	0.0455	0.0278	0.0468	0.0317
20-21	0.2500	0.2000	0.2708	0.2083	0.3256	0.2128	0.3977	0.3590	0.3205	0.7143	0.4407	0.3929	0.4251	0.3404
21-22	0.5000	0.3750	0.5455	0.4091	0.2453	0.1935	0.3559	0.1667	0.2769	0.5000	0.3409	0.3182	0.4474	0.4510
22-23	0.4000	0.1667	0.2500	0.1875	0.4583	0.2778	0.3077	0.6667	0.2917	0.4000	0.1515	0.0000	0.3196	0.3288
23-24	0.0000	0.5000	0.5833	0.3333	0.5238	0.1765	0.2258	0.0000	0.2857	0.4000	0.2143	0.5556	0.3377	0.1463
24-25	0.0000	0.2000	0.4444	0.5000	0.4545	0.2941	0.2174	0.1000	0.1818	0.0000	0.2222	0.0000	0.2083	0.2619
25-26	0.3333	0.0000	0.0000	0.3000	0.0000	0.0667	0.1250	0.2222	0.5000	0.5000	0.1538	0.0000	0.2609	0.2222
26-27	0.0000	0.2500	0.0000	0.3333	0.2500	0.5000	0.2000	0.1667	0.2000	0.0000	0.1429	0.0000	0.2308	0.1600
27-28	0.0000	0.0000	0.0000	0.5000	0.2222	0.3333	0.5000	0.5714	0.2000	1.0000	0.0000	0.0000	0.2381	0.4444
28-29	0.6000	0.0000	1.0000	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.5000	0.2000	0.0000	0.2222	0.0000
29-30	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.2500	0.0000	0.2500	0.0000	1.0000	0.0000	0.1765	0.2000
30-31	1.0000	1.0000	1.0000	0.6667	0.0000	0.7500	1.0000	0.8000	0.0000	0.0000	0.0000	0.0000	0.5385	0.2000

TABLE B-5 (Continued)

LOS &gt; 9

	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
9-10	0.0435	0.0818	0.0000	1.0000	0.1212	0.0609	0.0926	0.1063	0.1034	0.0455	0.0500	0.0526	0.1579	0.1481
10-11	0.0579	0.1584	0.1481	0.3333	0.1726	0.0708	0.1101	0.0980	0.0836	0.0426	0.0556	0.0500	0.0313	0.0345
11-12	0.0000	0.0936	0.0755	0.0000	0.1003	0.0702	0.0617	0.0843	0.1042	0.0789	0.0800	0.0667	0.1364	0.0541
12-13	0.0345	0.0805	0.0263	1.0000	0.0904	0.0566	0.0462	0.0724	0.0608	0.0313	0.0370	0.1579	0.0435	0.0400
13-14	0.0455	0.0722	0.0217	0.0000	0.0435	0.0488	0.0597	0.0544	0.0491	0.0571	0.0789	0.0000	0.0455	0.0619
14-15	0.0290	0.0233	0.1053	0.0000	0.0564	0.0448	0.0000	0.0515	0.0556	0.0571	0.0333	0.0588	0.0000	0.0265
15-16	0.0137	0.0455	0.0417	0.0000	0.0337	0.0346	0.0156	0.0232	0.0215	0.0667	0.0455	0.0000	0.0526	0.0174
16-17	0.0444	0.0451	0.0625	0.0000	0.0112	0.0658	0.0263	0.0215	0.0396	0.0000	0.0000	0.0000	0.1000	0.0000
17-18	0.0000	0.0254	0.0000	0.0000	0.0109	0.0325	0.0294	0.0181	0.0332	0.1071	0.0000	0.0000	0.0000	0.0088
18-19	0.0147	0.0143	0.0769	0.0000	0.0000	0.0187	0.0200	0.0172	0.0323	0.0000	0.0000	0.0000	0.0000	0.0000
19-20	0.0794	0.0403	0.0400	1.0000	0.0537	0.0442	0.0968	0.0658	0.0523	0.1053	0.1176	0.0000	0.1000	0.0345
20-21	0.4453	0.4298	0.3571	0.0000	0.2157	0.4444	0.3214	0.3684	0.3542	0.4615	0.3750	0.5000	0.4444	0.3176
21-22	0.2623	0.3030	0.1600	0.0000	0.3182	0.3077	0.2917	0.2977	0.2759	0.2857	0.1818	0.6000	0.3636	0.2828
22-23	0.3158	0.2885	0.2857	0.0000	0.2564	0.1667	0.3333	0.2547	0.2593	0.5000	0.0000	0.3333	0.0909	0.1277
23-24	0.3235	0.2329	0.1176	0.0000	0.2405	0.1707	0.3333	0.2000	0.1887	0.3750	0.3333	0.0000	0.3333	0.2903
24-25	0.2857	0.2833	0.0769	0.0000	0.1702	0.2683	0.0000	0.2353	0.0714	0.0000	0.1429	0.2500	0.3333	0.3214
25-26	0.2000	0.1944	0.0000	0.0000	0.2000	0.1915	0.5000	0.1279	0.1429	0.0000	0.2000	0.3333	0.1667	0.2000
26-27	0.5000	0.1818	0.2000	0.0000	0.2105	0.2632	0.3333	0.1818	0.1765	0.0000	0.0000	0.0000	0.4000	0.1250
27-28	0.2500	0.1818	0.0000	0.0000	0.0000	0.1667	0.0000	0.1765	0.2500	0.0000	0.0000	0.0000	0.3333	0.1429
28-29	0.0000	0.1000	0.5000	0.0000	0.0000	0.0000	0.0000	0.2258	0.1667	0.6667	0.5000	0.0000	0.0000	0.0000
29-30	0.0000	0.1250	0.0000	0.0000	0.0000	0.2500	0.0000	0.1374	0.0000	1.0000	0.0000	0.0000	0.0000	0.1667
30-31	1.0000	0.6250	1.0000	0.0000	0.7143	0.6000	0.0000	0.7200	0.8333	1.0000	0.0000	0.0000	1.0000	1.0000

TABLE B-5 (Continued)

LOS &gt; 9

	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
9-10	0.0000	0.0773	0.1176	0.0833	0.0705	0.0000	0.0000	0.1560	0.0938	0.1333	0.0938	0.0667	0.0000	0.1080
10-11	0.0000	0.1088	0.1250	0.1034	0.1039	0.0541	0.0405	0.1089	0.0484	0.0714	0.0156	0.1154	0.0000	0.0784
11-12	0.0000	0.0611	0.0000	0.0103	0.0438	0.0625	0.0494	0.0660	0.1486	0.0313	0.0588	0.1111	0.0000	0.0535
12-13	0.0000	0.0566	0.1000	0.0200	0.0519	0.0000	0.0267	0.0950	0.0244	0.0556	0.0217	0.0882	0.3333	0.0625
13-14	0.0000	0.0795	0.0000	0.0112	0.0320	0.0526	0.0100	0.0360	0.0208	0.0455	0.1429	0.0263	0.0000	0.0331
14-15	0.0000	0.0634	0.0000	0.0000	0.0381	0.0278	0.0179	0.0811	0.0465	0.0513	0.0286	0.0526	0.0000	0.0588
15-16	0.2000	0.0278	0.0789	0.0241	0.0297	0.0278	0.0248	0.0411	0.0784	0.0000	0.0000	0.0000	0.0000	0.0265
16-17	0.0000	0.0276	0.0000	0.0227	0.0107	0.0000	0.0090	0.0431	0.0000	0.0000	0.0000	0.0000	0.1667	0.0119
17-18	0.0000	0.0289	0.0000	0.0652	0.0042	0.0000	0.0000	0.0351	0.0000	0.0000	0.0465	0.0000	0.0000	0.0350
18-19	0.0000	0.0166	0.0000	0.0000	0.0064	0.0000	0.0102	0.0316	0.0000	0.0000	0.0000	0.0000	0.0000	0.0368
19-20	0.0000	0.0611	0.0833	0.0513	0.0304	0.0870	0.0610	0.1324	0.1333	0.1250	0.0250	0.0588	0.0000	0.0381
20-21	0.0000	0.3896	0.3333	0.4898	0.3219	0.3043	0.3553	0.4035	0.5000	0.1538	0.3125	0.5000	0.0000	0.2828
21-22	0.3333	0.2582	0.4000	0.4444	0.4193	0.1000	0.2118	0.3220	0.4706	0.4286	0.3077	0.4615	1.0000	0.2561
22-23	1.0000	0.2313	0.0000	0.5714	0.3421	0.2500	0.2885	0.2979	0.2500	0.0000	0.3333	0.3333	0.0000	0.3462
23-24	1.0000	0.3028	0.0000	0.2500	0.2571	0.1667	0.1875	0.2400	0.1250	0.2857	0.2500	0.0000	0.0000	0.2830
24-25	0.0000	0.2500	0.3333	0.2222	0.3377	0.1111	0.1228	0.2759	0.2857	0.2222	0.8000	0.5000	0.0000	0.1304
25-26	0.0000	0.2993	1.0000	0.0000	0.1826	0.0000	0.2045	0.2500	0.0000	0.1111	0.5000	0.3333	0.0000	0.2162
26-27	0.0000	0.0263	0.0000	0.3333	0.2500	0.3333	0.1563	0.2381	0.6667	0.0000	0.0000	0.3333	0.0000	0.2759
27-28	0.0000	0.1500	0.0000	0.3333	0.2364	0.3333	0.1053	0.0769	0.0000	1.0000	0.4286	0.2500	0.0000	0.1333
28-29	0.0000	0.0455	0.0000	0.0000	0.0513	0.3333	0.2667	0.0952	0.2000	0.0000	0.1667	0.0000	0.0000	0.1333
29-30	0.0000	0.0833	0.0000	0.0000	0.0769	0.0000	0.1765	0.1000	0.0000	0.0000	0.2500	0.0000	0.0000	0.6154
30-31	0.0000	0.5714	0.6667	0.3333	0.8462	0.8000	0.4286	0.7778	1.0000	0.0000	0.8571	0.0000	0.0000	0.6250

TABLE B-5 (Continued)

LOS &gt; 9

	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN
9-10	0.0175	0.1282	0.0986	0.0000	0.5000	0.1134	0.0451	0.1158	0.1915	0.4333	0.0844	0.0526	0.0963
10-11	0.0633	0.0703	0.0915	0.1667	0.5882	0.0797	0.0501	0.1327	0.1364	0.4063	0.0955	0.0920	0.0797
11-12	0.0513	0.0862	0.0585	0.1111	0.1667	0.0469	0.0311	0.0870	0.0746	0.2593	0.0301	0.0351	0.0502
12-13	0.0278	0.0313	0.0479	0.1333	0.6667	0.0333	0.0325	0.1139	0.0559	0.2632	0.0348	0.0290	0.0498
13-14	0.0822	0.0175	0.0430	0.0526	0.4000	0.0437	0.0422	0.0690	0.0704	0.1852	0.0897	0.0154	0.0597
14-15	0.0299	0.0609	0.0563	0.0357	0.3333	0.0608	0.0155	0.0423	0.0351	0.1250	0.0234	0.0508	0.0364
15-16	0.0159	0.0354	0.0316	0.0000	1.0000	0.0237	0.0186	0.0714	0.0286	0.0400	0.0490	0.0857	0.0260
16-17	0.0000	0.0390	0.0303	0.0435	1.0000	0.0201	0.0343	0.0645	0.0326	0.0625	0.0455	0.0247	0.0332
17-18	0.0238	0.0127	0.0215	0.0400	0.0000	0.0061	0.0290	0.0000	0.0238	0.0250	0.0135	0.0385	0.0426
18-19	0.0000	0.0400	0.0097	0.0435	0.0000	0.0052	0.0181	0.0000	0.0000	0.0000	0.0122	0.0175	0.0174
19-20	0.0357	0.0645	0.0223	0.0769	0.0000	0.0532	0.0227	0.0870	0.0513	0.1200	0.0962	0.1053	0.0532
20-21	0.3571	0.3333	0.4048	0.2143	0.0000	0.3196	0.3171	0.2692	0.3125	0.6923	0.4677	0.3939	0.3090
21-22	0.3571	0.3030	0.3211	0.3333	0.0000	0.3191	0.2488	0.1515	0.2264	0.5000	0.3636	0.3137	0.2480
22-23	0.3529	0.1250	0.2876	0.0000	0.0000	0.2973	0.3396	0.3846	0.2600	0.7500	0.3143	0.3030	0.2609
23-24	0.1429	0.2000	0.2844	0.3333	0.0000	0.3462	0.2877	0.0000	0.2000	0.8000	0.2903	0.2667	0.2740
24-25	0.2500	0.1200	0.2621	0.0000	0.0000	0.2609	0.2105	0.0000	0.2609	0.8000	0.2500	0.0000	0.2090
25-26	0.0000	0.2258	0.1176	0.3333	0.0000	0.0000	0.2162	0.1333	0.1333	0.0000	0.0833	0.1667	0.1385
26-27	0.2500	0.3077	0.1719	0.0000	0.0000	0.0000	0.2105	0.2273	0.2500	0.3333	0.2000	1.0000	0.2245
27-28	0.2500	0.1667	0.2439	0.0000	0.0000	0.1667	0.2857	0.0833	0.1429	0.0000	0.2500	0.2500	0.3103
28-29	0.2000	0.0000	0.1463	0.0000	0.0000	0.2000	0.1500	0.3333	0.0000	0.2500	0.1818	0.5000	0.1304
29-30	0.5000	0.3333	0.2162	0.0000	0.0000	0.1250	0.3333	0.1000	0.5000	1.0000	0.0000	0.0000	0.2667
30-31	0.2500	0.3333	0.5455	0.0000	0.0000	0.6364	0.6957	0.5000	0.5000	1.0000	0.1667	0.8333	0.8000

and LOS, in table B-4 are multiplied by the number of expected lateral transfers in the corresponding enlistment program and LOS to obtain the number of lateral transfers into each rating for each enlistment program and LOS. The number of expected lateral transfers in each enlistment program and LOS continuation is obtained from the ratio of total lateral transfers in the enlistment program and LOS to the number of individuals (in the begin year) in that enlistment program and LOS. Active duty loss rates by enlistment program, rating and LOS are in table B-5, as calculated from individual behavior.

#### Other Data

The paygrade distribution by rating and LOS tabulated for the September 1985 EMR appears in table B-6. Prior service accessions are allocated to rating groups and LOS (for LOS  $\leq$  9) according to the fractions in table B-7. The distribution is based on the September 1986 EMR allocation of Prior-Service accessions. Enlisted Programmed Authorizations (as of March 1986) are detailed, through FY 1991, in table B-8.

Table B-6

PAYGRADE DISTRIBUTION  
BY RATING AND LOS

7800 AN										6700 AB									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0335	0.0909	0.8756	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0121	0.0202	0.3090	0.6437	0.0135	0.0013	0.0000	0.0000	0.0000	
4	0.0529	0.0975	0.8496	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0104	0.0148	0.1346	0.6997	0.1376	0.0030	0.0000	0.0000	0.0000	
5	0.0611	0.0786	0.8603	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0058	0.0039	0.0425	0.6544	0.2915	0.0019	0.0000	0.0000	0.0000	
6	0.1233	0.0137	0.8630	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0041	0.0020	0.0348	0.5031	0.4438	0.0123	0.0000	0.0000	0.0000	
7	0.2121	0.1515	0.6364	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0083	0.0000	0.0331	0.3306	0.5537	0.0723	0.0021	0.0000	0.0000	
8	0.1875	0.0625	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0085	0.0021	0.0254	0.2754	0.5318	0.1568	0.0000	0.0000	0.0000	
9	0.0000	0.4444	0.5556	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0026	0.0000	0.0077	0.1748	0.4884	0.3265	0.0000	0.0000	0.0000	
10	0.5000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0036	0.0000	0.0145	0.0800	0.4182	0.4727	0.0109	0.0000	0.0000	
11	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0044	0.0796	0.2478	0.5619	0.1062	0.0000	0.0000	
12	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0085	0.0511	0.2255	0.6000	0.1149	0.0000	0.0000	
13	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0361	0.1649	0.5722	0.2268	0.0000	0.0000	
14	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0051	0.0000	0.0000	0.0253	0.1111	0.5000	0.3485	0.0101	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0046	0.0741	0.4074	0.5000	0.0139	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0971	0.4175	0.4466	0.0388	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0652	0.3913	0.4891	0.0543	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.3704	0.4815	0.0926	0.0185	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0244	0.3902	0.4634	0.1220	0.0000	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.2895	0.5000	0.1579	0.0000	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1471	0.5882	0.2647	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1176	0.5882	0.2353	0.0588	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0606	0.5152	0.3333	0.0909	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0652	0.3043	0.3696	0.2609	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0385	0.3077	0.3462	0.3077	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.2632	0.2105	0.4737	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1176	0.2353	0.2941	0.3529	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.4167	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.8182	

TABLE B-6 (Continued)

6600 AC										6080 AF									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0000	0.0215	0.6779	0.2822	0.0153	0.0031	0.0000	0.0000	0.0000	0.0093	0.0233	0.3211	0.6126	0.0333	0.0003	0.0000	0.0000	0.0000	
4	0.0000	0.0034	0.2248	0.6477	0.1174	0.0067	0.0000	0.0000	0.0000	0.0041	0.0136	0.1443	0.6248	0.2128	0.0004	0.0000	0.0000	0.0000	
5	0.0029	0.0000	0.0486	0.6686	0.2771	0.0029	0.0000	0.0000	0.0000	0.0058	0.0042	0.0530	0.4672	0.2678	0.0021	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0128	0.3319	0.6426	0.0128	0.0000	0.0000	0.0000	0.0022	0.0022	0.0282	0.3091	0.6393	0.0190	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0042	0.1345	0.7941	0.0630	0.0042	0.0000	0.0000	0.0014	0.0028	0.0152	0.1948	0.6924	0.0935	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0102	0.0914	0.6853	0.2132	0.0000	0.0000	0.0000	0.0026	0.0016	0.0181	0.1341	0.6023	0.2413	0.0000	0.0000	0.0000	
9	0.0051	0.0000	0.0000	0.0609	0.5736	0.3553	0.0051	0.0000	0.0000	0.0007	0.0007	0.0078	0.0361	0.4678	0.4240	0.0014	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0263	0.4145	0.5526	0.0056	0.0000	0.0000	0.0010	0.0010	0.0051	0.0449	0.3800	0.5577	0.0102	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0095	0.2000	0.7714	0.0190	0.0000	0.0000	0.0000	0.0000	0.0033	0.0354	0.2710	0.6416	0.0487	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.1169	0.8442	0.0390	0.0000	0.0000	0.0000	0.0000	0.0000	0.0307	0.2364	0.6537	0.0780	0.0012	0.0000	
13	0.0000	0.0000	0.0000	0.0101	0.0808	0.7273	0.1818	0.0000	0.0000	0.0000	0.0000	0.0024	0.0107	0.1603	0.7019	0.1235	0.0012	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0377	0.6981	0.2547	0.0094	0.0000	0.0000	0.0000	0.0000	0.0079	0.1240	0.6587	0.2063	0.0030	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0482	0.5542	0.3855	0.0120	0.0000	0.0012	0.0000	0.0000	0.0035	0.0631	0.6028	0.3166	0.0129	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0227	0.6591	0.2955	0.0227	0.0000	0.0000	0.0000	0.0000	0.0043	0.0541	0.5533	0.3400	0.0469	0.0014	
17	0.0000	0.0000	0.0000	0.0000	0.0152	0.5152	0.4545	0.0152	0.0000	0.0000	0.0000	0.0000	0.0014	0.0255	0.4837	0.4003	0.0877	0.0014	
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.3276	0.5172	0.1379	0.0172	0.0000	0.0000	0.0000	0.0000	0.0047	0.0204	0.3783	0.4003	0.1868	0.0094
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.3714	0.5714	0.0571	0.0000	0.0000	0.0000	0.0000	0.0000	0.0157	0.3933	0.3914	0.1781	0.0215	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.3793	0.2759	0.3103	0.0345	0.0000	0.0000	0.0000	0.0000	0.0159	0.3651	0.3783	0.2063	0.0344	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.4444	0.2222	0.0000	0.0000	0.0000	0.0000	0.0134	0.2785	0.3725	0.2651	0.0705	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.2222	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1878	0.3481	0.3315	0.1326	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6250	0.2500	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.1301	0.3973	0.3288	0.1438	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.2222	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0829	0.2798	0.4404	0.1969	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.6364	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.2857	0.3609	0.3008	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0233	0.2171	0.4109	0.3488	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.5385	0.2308	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1918	0.3014	0.5068	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.4000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1455	0.3273	0.5273	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.4000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1346	0.2692	0.5962	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0159	0.0317	0.4444	0.5079	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0227	0.1591	0.8182	

TABLE B-6 (Continued)

7100 AG										7300 AK									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0058	0.0117	0.1404	0.7018	0.1404	0.0000	0.0000	0.0000	0.0000	0.0069	0.0229	0.2700	0.6247	0.0755	0.0000	0.0000	0.0000	0.0000	
4	0.0000	0.0000	0.0438	0.4745	0.4818	0.0000	0.0000	0.0000	0.0000	0.0022	0.0155	0.1552	0.5809	0.2439	0.0022	0.0000	0.0000	0.0000	
5	0.0000	0.0000	0.0091	0.3455	0.6273	0.0182	0.0000	0.0000	0.0000	0.0024	0.0072	0.0553	0.4639	0.4688	0.0000	0.0000	0.0000	0.0024	
6	0.0095	0.0000	0.0190	0.1333	0.7714	0.0667	0.0000	0.0000	0.0000	0.0000	0.0026	0.0281	0.3581	0.6010	0.0102	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0108	0.1290	0.5269	0.3333	0.0000	0.0000	0.0000	0.0026	0.0000	0.0183	0.2304	0.7173	0.0314	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0111	0.1111	0.4000	0.4778	0.0000	0.0000	0.0000	0.0036	0.0000	0.0249	0.2171	0.6726	0.0619	0.0000	0.0000	0.0000	
9	0.0000	0.0000	0.0152	0.0152	0.3182	0.6515	0.0000	0.0000	0.0000	0.0037	0.0037	0.0037	0.0701	0.6937	0.2251	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0423	0.2254	0.7183	0.0141	0.0000	0.0000	0.0000	0.0046	0.0092	0.0688	0.5734	0.3394	0.0046	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0000	0.1795	0.6923	0.1282	0.0000	0.0000	0.0000	0.0000	0.0056	0.0556	0.5333	0.3778	0.0278	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.1190	0.6429	0.2381	0.0000	0.0000	0.0000	0.0000	0.0000	0.0488	0.4695	0.4268	0.0549	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.0789	0.7368	0.1842	0.0000	0.0000	0.0000	0.0000	0.0122	0.0122	0.3354	0.5549	0.0854	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0638	0.6383	0.2979	0.0000	0.0000	0.0000	0.0000	0.0053	0.0213	0.1755	0.6649	0.1330	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0286	0.5714	0.3714	0.0286	0.0000	0.0000	0.0000	0.0071	0.0071	0.1429	0.6214	0.2071	0.0143	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0114	0.1477	0.4773	0.3523	0.0114	0.0000	
17	0.0000	0.0000	0.0000	0.0476	0.0476	0.4286	0.4286	0.0476	0.0000	0.0000	0.0000	0.0000	0.0244	0.0366	0.4512	0.3780	0.0976	0.0122	
18	0.0000	0.0000	0.0000	0.0000	0.0333	0.3667	0.5333	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0455	0.5682	0.3409	0.0455	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.2759	0.5172	0.2069	0.0000	0.0000	0.0000	0.0000	0.0000	0.0265	0.4513	0.3982	0.1150	0.0088	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.4375	0.4375	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0494	0.4568	0.3457	0.1235	0.0247	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.3158	0.4211	0.1579	0.1053	0.0000	0.0000	0.0000	0.0000	0.0000	0.4659	0.3864	0.1250	0.0227	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.3000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4615	0.3846	0.0769	0.0769	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.3000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3478	0.3043	0.1304	0.2174	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.7273	0.0909	0.0000	0.0000	0.0000	0.0000	0.0476	0.0952	0.4762	0.1429	0.2381	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.2857	0.0000	0.4286	0.0000	0.0000	0.0000	0.0000	0.0667	0.2000	0.2667	0.2000	0.2000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.4167	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.3125	0.1875	0.4375	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.3333	0.4167	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.1429	0.4286	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.3750	0.3750	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.1250	0.2500	0.5000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	



TABLE B-6 (Continued)

6500 AO										7500 AS									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0139	0.0254	0.3075	0.0081	0.0439	0.0012	0.0000	0.0000	0.0000	0.0055	0.0110	0.2590	0.6777	0.0441	0.0028	0.0000	0.0000	0.0000	
4	0.0087	0.0137	0.1021	0.6438	0.2304	0.0012	0.0000	0.0000	0.0000	0.0091	0.0061	0.1003	0.6413	0.2432	0.0000	0.0000	0.0000	0.0000	
5	0.0032	0.0032	0.0427	0.4462	0.5032	0.0016	0.0000	0.0000	0.0000	0.0084	0.0127	0.0422	0.4641	0.4726	0.0000	0.0000	0.0000	0.0000	
6	0.0000	0.0065	0.0323	0.2866	0.6401	0.0323	0.0000	0.0000	0.0022	0.0000	0.0000	0.0144	0.2823	0.6938	0.0096	0.0000	0.0000	0.0000	
7	0.0044	0.0022	0.0131	0.1488	0.6871	0.1444	0.0000	0.0000	0.0000	0.0000	0.0076	0.0076	0.1582	0.7704	0.0561	0.0000	0.0000	0.0000	
8	0.0023	0.0023	0.0094	0.1077	0.5457	0.3326	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0619	0.5575	0.3805	0.0000	0.0000	0.0000	
9	0.0027	0.0000	0.0000	0.0676	0.4243	0.5000	0.0054	0.0000	0.0000	0.0000	0.0000	0.0116	0.0698	0.5698	0.3488	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0042	0.0466	0.2542	0.6695	0.0254	0.0000	0.0000	0.0000	0.0000	0.0000	0.0484	0.3871	0.5484	0.0161	0.0000	0.0000	
11	0.0000	0.0051	0.0102	0.0357	0.1633	0.7245	0.0612	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0114	0.1136	0.7841	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.0448	0.4030	0.5224	0.0299	0.0000	0.0000	
13	0.0000	0.0000	0.0114	0.0227	0.0739	0.6705	0.2216	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2791	0.6279	0.0814	0.0116	0.0000	
14	0.0000	0.0000	0.0044	0.0089	0.0489	0.6667	0.2622	0.0044	0.0044	0.0000	0.0000	0.0000	0.0093	0.2130	0.6111	0.1667	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0222	0.6074	0.3556	0.0148	0.0000	0.0000	0.0000	0.0141	0.0282	0.1127	0.5634	0.2817	0.0000	0.0000	
16	0.0093	0.0000	0.0000	0.0000	0.0280	0.5701	0.3645	0.0280	0.0000	0.0000	0.0000	0.0000	0.0141	0.1268	0.6056	0.2254	0.0282	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0085	0.5085	0.3729	0.1102	0.0000	0.0000	0.0000	0.0000	0.0000	0.0811	0.5676	0.2973	0.0541	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0102	0.4082	0.4694	0.0918	0.0204	0.0000	0.0000	0.0000	0.0122	0.0366	0.5488	0.2317	0.1707	0.0000	
19	0.0000	0.0000	0.0000	0.0105	0.0105	0.4316	0.4000	0.1368	0.0105	0.0000	0.0000	0.0000	0.0000	0.0357	0.6250	0.2143	0.0893	0.0357	
20	0.0000	0.0000	0.0000	0.0000	0.0145	0.4058	0.3768	0.1449	0.0580	0.0000	0.0000	0.0000	0.0000	0.0250	0.4250	0.1750	0.2750	0.1000	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.4318	0.2500	0.0455	0.0000	0.0000	0.0000	0.0000	0.0000	0.4800	0.3600	0.1200	0.0400	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.2333	0.4333	0.2333	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4500	0.2500	0.3000	0.0000	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.2121	0.5152	0.0909	0.1818	0.0000	0.0000	0.0000	0.0000	0.1250	0.0000	0.2500	0.5000	0.1250	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.1707	0.2927	0.3171	0.2195	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.1250	0.3750	0.2500	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0714	0.4286	0.1429	0.3571	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.1429	0.5714	0.1429	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.2000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.0000	0.4000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2857	0.4286	0.1429	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.2500	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.1250	0.2500	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2000	0.4000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.2000	0.0000	0.7000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.0000	0.8889	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	

TABLE B-6 (Continued)

6180 AV										6400 AW									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0000
3	0.0053	0.0198	0.4355	0.4257	0.1137	0.0000	0.0000	0.0000	0.0000	0.0000	0.0078	0.2000	0.6883	0.1039	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0015	0.0161	0.2652	0.4130	0.3042	0.0000	0.0000	0.0000	0.0000	0.0027	0.0054	0.0536	0.2949	0.6434	0.0000	0.0000	0.0000	0.0000	0.0000
5	0.0005	0.0050	0.0864	0.3327	0.5744	0.0010	0.0000	0.0000	0.0000	0.0035	0.0035	0.0141	0.1060	0.8587	0.0141	0.0000	0.0000	0.0000	0.0000
6	0.0005	0.0035	0.0319	0.1882	0.7495	0.0264	0.0000	0.0000	0.0000	0.0000	0.0000	0.0099	0.1040	0.8267	0.0594	0.0000	0.0000	0.0000	0.0000
7	0.0018	0.0012	0.0259	0.1179	0.7506	0.1026	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0621	0.6211	0.3168	0.0000	0.0000	0.0000	0.0000
8	0.0007	0.0020	0.0119	0.0871	0.6403	0.2568	0.0013	0.0000	0.0000	0.0000	0.0000	0.0042	0.0338	0.5443	0.4135	0.0000	0.0042	0.0000	0.0000
9	0.0000	0.0000	0.0079	0.0525	0.5517	0.3844	0.0035	0.0000	0.0000	0.0000	0.0000	0.0000	0.0158	0.4000	0.5842	0.0000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0032	0.0324	0.4391	0.5102	0.0151	0.0000	0.0000	0.0000	0.0000	0.0060	0.0060	0.2289	0.7349	0.0000	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0022	0.0221	0.2928	0.6144	0.0685	0.0000	0.0000	0.0000	0.0000	0.0000	0.0071	0.1143	0.7571	0.1214	0.0000	0.0000	0.0000
12	0.0000	0.0000	0.0027	0.0190	0.2520	0.6260	0.1003	0.0000	0.0000	0.0000	0.0000	0.0000	0.0099	0.0693	0.8218	0.0990	0.0000	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0093	0.1611	0.6644	0.1611	0.0040	0.0000	0.0000	0.0000	0.0000	0.0000	0.0291	0.6796	0.2913	0.0000	0.0000	0.0000
14	0.0000	0.0000	0.0013	0.0121	0.1060	0.6792	0.1919	0.0094	0.0000	0.0000	0.0000	0.0000	0.0000	0.0230	0.7011	0.2759	0.0000	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0016	0.0916	0.5766	0.3144	0.0142	0.0016	0.0000	0.0000	0.0000	0.0000	0.0241	0.4940	0.4819	0.0000	0.0000	0.0000
16	0.0000	0.0000	0.0015	0.0015	0.0573	0.5400	0.3484	0.0513	0.0000	0.0000	0.0000	0.0000	0.0000	0.0159	0.5714	0.3492	0.0317	0.0317	0.0317
17	0.0000	0.0000	0.0000	0.0000	0.0377	0.4705	0.3607	0.1262	0.0049	0.0000	0.0000	0.0000	0.0000	0.0169	0.3729	0.4746	0.1186	0.0169	0.0169
18	0.0000	0.0000	0.0000	0.0000	0.0185	0.4465	0.3601	0.1584	0.0165	0.0000	0.0000	0.0000	0.0000	0.0000	0.2581	0.7097	0.0323	0.0000	0.0000
19	0.0000	0.0000	0.0000	0.0022	0.0217	0.3435	0.3891	0.2174	0.0261	0.0000	0.0000	0.0000	0.0000	0.0323	0.3226	0.5161	0.1290	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0165	0.3102	0.3168	0.2904	0.0660	0.0000	0.0000	0.0000	0.0000	0.0000	0.1212	0.5455	0.3333	0.0000	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1635	0.3802	0.3612	0.0951	0.0000	0.0000	0.0000	0.0000	0.0455	0.1818	0.3636	0.3182	0.0909	0.0909
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1786	0.2440	0.3750	0.2024	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.4000	0.0000	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0811	0.2703	0.5045	0.1441	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4500	0.5000	0.0000	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0619	0.2301	0.4867	0.2212	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.3158	0.5263	0.1053	0.1053
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0141	0.1549	0.4648	0.3662	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7000	0.2000	0.1000	0.1000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1970	0.4394	0.3636	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3571	0.2143	0.4286	0.4286
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.4000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.3750	0.5000	0.5000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0149	0.1343	0.3582	0.4925	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.1250	0.3750	0.3750
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1200	0.2600	0.6200	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.5000	0.2500
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0606	0.3939	0.5455	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0400	0.2800	0.6800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.0000	0.5000

TABLE B-6 (Continued)

7400 AZ												0100 BM											
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9					
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000					
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000					
3	0.0000	0.0000	0.1589	0.7042	0.1280	0.0000	0.0000	0.0000	0.0000	0.0022	0.0111	0.1889	0.7544	0.0400	0.0022	0.0011	0.0000	0.0000					
4	0.0045	0.0000	0.0633	0.5588	0.3543	0.0000	0.0000	0.0000	0.0000	0.0038	0.0107	0.1222	0.6738	0.1849	0.0031	0.0015	0.0000	0.0000					
5	0.0000	0.0056	0.0501	0.3816	0.5571	0.0028	0.0000	0.0028	0.0000	0.0010	0.0096	0.0674	0.5659	0.3494	0.0058	0.0010	0.0000	0.0000					
6	0.0000	0.0000	0.0104	0.2396	0.7049	0.0451	0.0000	0.0000	0.0000	0.0054	0.0068	0.0486	0.3811	0.5365	0.0189	0.0027	0.0000	0.0000					
7	0.0000	0.0000	0.0207	0.1612	0.7107	0.1074	0.0000	0.0000	0.0000	0.0015	0.0030	0.0378	0.2583	0.6133	0.0816	0.0045	0.0000	0.0000					
8	0.0000	0.0000	0.0000	0.1136	0.6636	0.2227	0.0000	0.0000	0.0000	0.0047	0.0047	0.0267	0.2091	0.5645	0.1871	0.0031	0.0000	0.0000					
9	0.0061	0.0000	0.0061	0.0424	0.5515	0.3879	0.0061	0.0000	0.0000	0.0017	0.0035	0.0157	0.1469	0.4808	0.3444	0.0070	0.0000	0.0000					
10	0.0000	0.0000	0.0000	0.0513	0.5043	0.4444	0.0000	0.0000	0.0000	0.0000	0.0026	0.0104	0.0933	0.3497	0.5311	0.0130	0.0000	0.0000					
11	0.0000	0.0000	0.0000	0.0196	0.3039	0.6275	0.0490	0.0000	0.0000	0.0025	0.0000	0.0050	0.0501	0.3233	0.5514	0.0652	0.0025	0.0000					
12	0.0000	0.0000	0.0000	0.0083	0.2893	0.6281	0.0744	0.0000	0.0000	0.0000	0.0000	0.0028	0.0497	0.2293	0.5856	0.1271	0.0055	0.0000					
13	0.0000	0.0000	0.0000	0.0315	0.2047	0.6535	0.1102	0.0000	0.0000	0.0022	0.0000	0.0044	0.0264	0.1608	0.5815	0.2247	0.0000	0.0000					
14	0.0000	0.0000	0.0000	0.0085	0.1368	0.6410	0.2137	0.0000	0.0000	0.0000	0.0000	0.0000	0.0195	0.1098	0.5805	0.2829	0.0073	0.0000					
15	0.0081	0.0000	0.0000	0.0163	0.1138	0.6260	0.2358	0.0000	0.0000	0.0000	0.0000	0.0000	0.0045	0.0495	0.4189	0.4865	0.0405	0.0000					
16	0.0000	0.0000	0.0000	0.0000	0.0500	0.6625	0.2750	0.0125	0.0000	0.0000	0.0000	0.0000	0.0037	0.0407	0.3815	0.4333	0.1370	0.0037					
17	0.0000	0.0000	0.0000	0.0000	0.0533	0.5467	0.3467	0.0533	0.0000	0.0000	0.0000	0.0000	0.0000	0.0465	0.3547	0.4244	0.1686	0.0000					
18	0.0000	0.0000	0.0000	0.0000	0.0270	0.5000	0.3919	0.0811	0.0000	0.0058	0.0000	0.0000	0.0000	0.0566	0.2453	0.5346	0.1572	0.0063					
19	0.0000	0.0000	0.0000	0.0000	0.0286	0.4429	0.4286	0.0857	0.0143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
20	0.0000	0.0000	0.0000	0.0000	0.0517	0.3793	0.4483	0.0862	0.0345	0.0000	0.0000	0.0000	0.0000	0.0173	0.3237	0.4451	0.1734	0.0405					
21	0.0000	0.0000	0.0000	0.0000	0.0179	0.3214	0.4821	0.1071	0.0714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1714	0.6571	0.1143	0.0571	0.0000	0.0000	0.0000	0.0000	0.0085	0.1186	0.4661	0.3136	0.0932					
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.3500	0.2000	0.2500	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.4167	0.1250	0.2917	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.1538	0.1538	0.0769	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000					
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0263	0.1316	0.3158	0.5263					
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000	0.2000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0417	0.0000	0.1667	0.3333	0.4583					
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0.0588	0.1176	0.7647					
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0.1765	0.4706	0.2941					
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.7273					

TABLE B-6 (Continued)

4000 BT											6000 CN										
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9		PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9		
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000		0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000		
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000		0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000		
3	0.0122	0.0479	0.3320	0.5446	0.0633	0.0000	0.0000	0.0000	0.0000		0.3333	0.0833	0.5833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
4	0.0120	0.0248	0.1827	0.5938	0.1867	0.0000	0.0000	0.0000	0.0000		1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
5	0.0086	0.0148	0.0701	0.4526	0.4428	0.0098	0.0012	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
6	0.0027	0.0067	0.0373	0.3107	0.5187	0.1240	0.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
7	0.0073	0.0049	0.0280	0.2549	0.5183	0.1854	0.0012	0.0000	0.0000		0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
8	0.0016	0.0032	0.0190	0.1965	0.4200	0.3597	0.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
9	0.0097	0.0019	0.0174	0.1143	0.3547	0.4942	0.0078	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
10	0.0023	0.0000	0.0046	0.0691	0.2834	0.5945	0.0461	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
11	0.0026	0.0026	0.0000	0.0396	0.1636	0.5989	0.1926	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
12	0.0031	0.0000	0.0000	0.0245	0.1590	0.6268	0.1927	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
13	0.0000	0.0000	0.0070	0.0211	0.1338	0.4507	0.3838	0.0000	0.0035		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
14	0.0000	0.0000	0.0000	0.0278	0.0635	0.4286	0.4722	0.0079	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
15	0.0000	0.0000	0.0000	0.0162	0.0595	0.3081	0.5405	0.0757	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
16	0.0000	0.0000	0.0000	0.0159	0.0397	0.2460	0.5794	0.1111	0.0079		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
17	0.0079	0.0000	0.0000	0.0236	0.0472	0.3071	0.4094	0.2047	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
18	0.0000	0.0000	0.0000	0.0101	0.0303	0.2222	0.3434	0.3131	0.0808		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
19	0.0000	0.0000	0.0000	0.0278	0.0278	0.1250	0.3889	0.3194	0.1111		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
20	0.0000	0.0000	0.0000	0.0000	0.0485	0.2039	0.2913	0.3301	0.1262		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
21	0.0000	0.0000	0.0000	0.0000	0.0104	0.1042	0.3646	0.3646	0.1563		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0536	0.2857	0.2679	0.3929		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0278	0.1389	0.3333	0.5000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.1852	0.3333	0.4444		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2368	0.1842	0.5789		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
26	0.0435	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1739	0.6087		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.0833	0.8333		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.1000	0.8000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		

TABLE B-6 (Continued)

1622 CTA										1666 CTI									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0000	0.0000	0.1458	0.7396	0.1146	0.0000	0.0000	0.0000	0.0000	0.0000	0.0214	0.1357	0.7643	0.0786	0.0000	0.0000	0.0000	0.0000	
4	0.0000	0.0000	0.0450	0.4414	0.5135	0.0000	0.0000	0.0000	0.0000	0.0000	0.0101	0.0303	0.3838	0.5758	0.0000	0.0000	0.0000	0.0000	
5	0.0000	0.0000	0.0000	0.3049	0.6829	0.0122	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0098	0.1961	0.7941	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0000	0.1000	0.9000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0824	0.9176	0.0000	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0147	0.1029	0.8235	0.0588	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0357	0.7500	0.2143	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0000	0.0143	0.6714	0.3143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0156	0.0625	0.3125	0.6094	0.0000	0.0000	0.0000	
9	0.0000	0.0000	0.0127	0.0000	0.4684	0.5190	0.0000	0.0000	0.0000	0.0000	0.0154	0.0000	0.0154	0.2769	0.6923	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0185	0.3889	0.5926	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0794	0.8254	0.0952	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0370	0.2037	0.7037	0.0556	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1613	0.8226	0.0161	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.2340	0.6596	0.1064	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0811	0.6216	0.2973	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.0732	0.5610	0.3659	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0476	0.7619	0.1905	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0811	0.5946	0.3243	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.2917	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0667	0.4667	0.4000	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.5652	0.3913	0.0000	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7143	0.9357	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.4074	0.5556	0.0000	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0625	0.3750	0.4688	0.0938	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3871	0.5161	0.0968	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0769	0.3077	0.3846	0.1538	0.0769	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.5000	0.0714	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0526	0.1579	0.4211	0.3684	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2381	0.5714	0.1905	0.0000	
20	0.0000	0.0000	0.0000	0.0000	0.0833	0.0833	0.1667	0.4167	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.2667	0.0667	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.2500	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.3750	0.3750	0.2500	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.3000	0.3000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.1667	0.1667	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.6000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.2500	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2000	0.4000	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.3333	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.0000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.2500	0.5000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

1633 CTM

1644 CTO

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0000	0.0000	0.0079	0.9444	0.0476	0.0000	0.0000	0.0000	0.0000	0.0036	0.0000	0.3066	0.6460	0.0438	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0046	0.7685	0.2269	0.0000	0.0000	0.0000	0.0000	0.0076	0.0076	0.0916	0.6221	0.2710	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.4188	0.5812	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0149	0.4627	0.5224	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0040	0.0000	0.1862	0.8097	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.2735	0.6752	0.0256	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.1060	0.8609	0.0331	0.0000	0.0000	0.0000	0.0000	0.0090	0.0000	0.2252	0.7117	0.0541	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0382	0.7863	0.1756	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1490	0.6560	0.2000	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0444	0.5704	0.3852	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0792	0.4950	0.4257	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0276	0.5379	0.4207	0.0138	0.0000	0.0000	0.0000	0.0000	0.0217	0.0652	0.3261	0.5870	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0165	0.2231	0.7355	0.0248	0.0000	0.0000	0.0000	0.0000	0.0000	0.0482	0.2771	0.6627	0.0120	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.2264	0.6604	0.1132	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2292	0.5667	0.1042	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0164	0.1639	0.6393	0.1639	0.0164	0.0000	0.0000	0.0000	0.0000	0.0154	0.1692	0.6154	0.2000	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0000	0.6923	0.1923	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0682	0.7273	0.2045	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0645	0.5161	0.4194	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4681	0.4468	0.0213	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.5385	0.4583	0.0833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0408	0.6122	0.3061	0.0408	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.3469	0.4583	0.0833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0233	0.4884	0.4186	0.0698	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.3019	0.5283	0.1321	0.0189	0.0000	0.0000	0.0000	0.0000	0.0000	0.3659	0.3902	0.2195	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.0189	0.3019	0.5283	0.1321	0.0189	0.0000	0.0000	0.0000	0.0000	0.0000	0.3462	0.2692	0.3846	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.2708	0.4167	0.2292	0.0833	0.0000	0.0000	0.0000	0.0000	0.0000	0.4167	0.3333	0.1250	0.1250
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.4545	0.2727	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3636	0.2727	0.3182	0.0455
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.3125	0.1875	0.3125	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.4375	0.2500	0.1875
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.3333	0.2500	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8889	0.0000	0.1111
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.3000	0.3000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667



TABLE B-6 (Continued)

1655 CTR										1611 CTT									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0000	0.0036	0.2190	0.7153	0.0620	0.0000	0.0000	0.0000	0.0000	0.0000	0.0078	0.1783	0.7674	0.0465	0.0000	0.0000	0.0000	0.0000	
4	0.0000	0.0000	0.0379	0.6635	0.2938	0.0047	0.0000	0.0000	0.0000	0.0000	0.0041	0.0366	0.4512	0.5081	0.0000	0.0000	0.0000	0.0000	
5	0.0068	0.0000	0.0338	0.3378	0.6216	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0144	0.1587	0.8173	0.0096	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0000	0.2784	0.7113	0.0103	0.0000	0.0000	0.0000	0.0000	0.0000	0.0139	0.1111	0.8056	0.0694	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0095	0.0571	0.7810	0.1524	0.0000	0.0000	0.0000	0.0000	0.0000	0.0061	0.0485	0.6242	0.3212	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0000	0.0395	0.6184	0.3421	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.4583	0.4917	0.0000	0.0000	0.0000	
9	0.0000	0.0000	0.0000	0.0417	0.4271	0.5208	0.0104	0.0000	0.0000	0.0000	0.0000	0.0000	0.0102	0.2143	0.7653	0.0102	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0000	0.3067	0.6933	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0090	0.1081	0.8559	0.0270	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0000	0.1034	0.8851	0.0115	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0741	0.8272	0.0988	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7971	0.1159	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.7179	0.2821	0.0000	0.0000	0.0000	0.0000	0.0000	0.0145	0.0725	0.7971	0.1159	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0333	0.6667	0.3000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0313	0.0000	0.4375	0.5313	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0213	0.4255	0.5532	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6735	0.3061	0.0204	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0213	0.5319	0.4468	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5476	0.4286	0.0238	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.5600	0.1400	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.4286	0.0714	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0370	0.3333	0.5556	0.0741	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3250	0.6000	0.0750	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.3158	0.5789	0.1053	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3243	0.4865	0.1892	0.0000	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.3256	0.4186	0.2093	0.0465	0.0000	0.0000	0.0000	0.0000	0.0000	0.1915	0.5106	0.2553	0.0426	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1887	0.4528	0.3019	0.0566	0.0000	0.0000	0.0000	0.0000	0.0000	0.2581	0.4516	0.2581	0.0323	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.6250	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.5556	0.1667	0.1667	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.2381	0.4286	0.2857	0.0476	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0.6471	0.1176	0.1765	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0909	0.3636	0.2727	0.2727	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0.5294	0.2941	0.1176	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.2667	0.4000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.3333	0.2222	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.7500	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	

TABLE B-6 (Continued)

5000 CU

8300 DN

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0078	0.0352	0.4736	0.4286	0.0489	0.0059	0.0000	0.0000	0.0000	0.0056	0.0112	0.6927	0.2849	0.0056	0.0000	0.0000	0.0000	0.0000
4	0.0144	0.0263	0.3301	0.4187	0.2057	0.0048	0.0000	0.0000	0.0000	0.0031	0.0122	0.4220	0.5352	0.0275	0.0000	0.0000	0.0000	0.0000
5	0.0068	0.0137	0.1675	0.4188	0.3829	0.0103	0.0000	0.0000	0.0000	0.0000	0.0028	0.2444	0.6348	0.1152	0.0028	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0538	0.3259	0.5854	0.0348	0.0000	0.0000	0.0000	0.0000	0.0000	0.1026	0.6513	0.2462	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0482	0.1847	0.6345	0.1325	0.0000	0.0000	0.0000	0.0000	0.0000	0.0611	0.4978	0.4279	0.0131	0.0000	0.0000	0.0000
8	0.0000	0.0044	0.0175	0.1223	0.5415	0.3144	0.0000	0.0000	0.0000	0.0000	0.0000	0.0350	0.3813	0.5409	0.0428	0.0000	0.0000	0.0000
9	0.0000	0.0062	0.0123	0.0802	0.4321	0.4568	0.0123	0.0000	0.0000	0.0000	0.0000	0.0065	0.2323	0.6452	0.1161	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0164	0.0246	0.4098	0.5492	0.0000	0.0000	0.0000	0.0000	0.0000	0.0092	0.1193	0.5963	0.2661	0.0000	0.0092	0.0000
11	0.0000	0.0141	0.0000	0.0423	0.2113	0.6761	0.0563	0.0000	0.0000	0.0000	0.0000	0.0105	0.0947	0.4211	0.4211	0.0526	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0110	0.1319	0.6374	0.2198	0.0000	0.0000	0.0000	0.0000	0.0000	0.0694	0.4167	0.4444	0.0694	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0130	0.1429	0.6623	0.1818	0.0000	0.0000	0.0102	0.0000	0.0000	0.0102	0.2857	0.6020	0.0918	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0423	0.6761	0.2676	0.0141	0.0000	0.0000	0.0000	0.0000	0.0213	0.1489	0.5745	0.2553	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0094	0.0283	0.5849	0.3679	0.0094	0.0000	0.0000	0.0000	0.0000	0.0154	0.1385	0.5692	0.2615	0.0154	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0381	0.5238	0.4190	0.0190	0.0000	0.0000	0.0000	0.0000	0.0000	0.1045	0.5522	0.2985	0.0448	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0440	0.5714	0.3187	0.0659	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.4815	0.4444	0.0370	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0380	0.3797	0.3671	0.2152	0.0000	0.0000	0.0000	0.0000	0.0000	0.0172	0.5862	0.3276	0.0517	0.0172
19	0.0000	0.0000	0.0000	0.0000	0.0313	0.4063	0.3906	0.1563	0.0156	0.0000	0.0000	0.0000	0.0000	0.0000	0.3684	0.5000	0.1053	0.0263
20	0.0000	0.0000	0.0000	0.0000	0.0227	0.3750	0.3409	0.2045	0.0568	0.0000	0.0000	0.0256	0.0000	0.0000	0.4103	0.2821	0.2564	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.3051	0.4407	0.2203	0.0339	0.0000	0.0000	0.0000	0.0000	0.0000	0.5556	0.2778	0.1667	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.2051	0.4359	0.2564	0.1026	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.4444	0.1111	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0645	0.5161	0.3226	0.0968	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.4167	0.4167	0.0833
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.1739	0.4348	0.3478	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.1000	0.4000	0.4000
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.1250	0.3750	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.2222	0.4444	0.2222
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.0000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.2857	0.2857
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.0000	0.5714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



TABLE B-6 (Continued)

2100 DK

3200 DM

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0000	0.0134	0.2450	0.6342	0.1040	0.0000	0.0000	0.0034	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0058	0.0638	0.6174	0.3101	0.0029	0.0000	0.0000	0.0000	0.0000	0.0000	0.0357	0.7500	0.2143	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0251	0.4922	0.4828	0.0000	0.0000	0.0000	0.0000	0.0000	0.0286	0.4000	0.5429	0.0286	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0328	0.2372	0.7153	0.0146	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4333	0.5667	0.0000	0.0000	0.0000	0.0000
7	0.0052	0.0000	0.0157	0.1361	0.7277	0.1152	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1600	0.8000	0.0400	0.0000	0.0000	0.0000
8	0.0093	0.0000	0.0093	0.1667	0.5278	0.2870	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1081	0.7568	0.1351	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0500	0.4333	0.5167	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.6154	0.2692	0.0000	0.0000	0.0000
10	0.0128	0.0000	0.0000	0.0769	0.2564	0.6538	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.6522	0.3043	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.7931	0.0230	0.0000	0.0000	0.0000	0.0000	0.0000	0.0714	0.4286	0.5000	0.0000	0.0000	0.0000
12	0.0093	0.0000	0.0000	0.0093	0.1121	0.8131	0.0561	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.5000	0.4000	0.0500	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.1091	0.7818	0.1000	0.0091	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.8333	0.0833	0.0000	0.0000
14	0.0179	0.0000	0.0000	0.0000	0.0536	0.8393	0.0893	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0488	0.7073	0.2195	0.0244	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.6000	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0400	0.6800	0.2800	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7857	0.1429	0.0714
17	0.0000	0.0000	0.0000	0.0139	0.0278	0.7083	0.1944	0.0556	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2857	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0111	0.6556	0.2889	0.0444	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.5714	0.2857	0.0000	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.6517	0.2360	0.1124	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	0.0000	0.0000
20	0.0000	0.0000	0.0128	0.0000	0.0000	0.6538	0.1795	0.1410	0.0128	0.0000	0.0000	0.0000	0.0000	0.0000	0.5714	0.4286	0.0000	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.3538	0.1846	0.0615	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.2500	0.2500	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0417	0.2500	0.2500	0.4583	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.4286	0.1429	0.2857	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.3636	0.3636	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.5000
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.2000	0.5000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.5000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	0.0000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.2857	0.2857	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-6 (Continued)

1900 DP										1910 DS									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0020	0.0184	0.6184	0.3122	0.0490	0.0000	0.0000	0.0000	0.0000	0.0000	0.0027	0.0429	0.9115	0.0429	0.0000	0.0000	0.0000	0.0000	
4	0.0035	0.0088	0.3866	0.4095	0.1916	0.0000	0.0000	0.0000	0.0000	0.0000	0.0055	0.0443	0.7701	0.1801	0.0000	0.0000	0.0000	0.0000	
5	0.0016	0.0016	0.1947	0.4252	0.3738	0.0031	0.0000	0.0000	0.0000	0.0000	0.0065	0.0645	0.5000	0.4290	0.0000	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0477	0.3484	0.5418	0.0621	0.0000	0.0000	0.0000	0.0000	0.0000	0.0388	0.2746	0.6746	0.0119	0.0000	0.0000	0.0000	
7	0.0000	0.0033	0.0423	0.2280	0.4853	0.2410	0.0000	0.0000	0.0000	0.0049	0.0000	0.0098	0.0784	0.5784	0.3284	0.0000	0.0000	0.0000	
8	0.0039	0.0000	0.0154	0.2085	0.4595	0.3127	0.0000	0.0000	0.0000	0.0000	0.0000	0.0092	0.0415	0.4009	0.5346	0.0138	0.0000	0.0000	
9	0.0000	0.0000	0.0190	0.1232	0.2986	0.5403	0.0190	0.0000	0.0000	0.0000	0.0000	0.0068	0.0405	0.2500	0.6419	0.0608	0.0000	0.0000	
10	0.0000	0.0000	0.0061	0.0915	0.2561	0.6220	0.0244	0.0000	0.0000	0.0000	0.0000	0.0000	0.0181	0.1386	0.7169	0.1265	0.0000	0.0000	
11	0.0000	0.0000	0.0083	0.0496	0.2149	0.6446	0.0826	0.0000	0.0000	0.0000	0.0000	0.0000	0.0144	0.1223	0.6619	0.2014	0.0000	0.0000	
12	0.0000	0.0000	0.168	0.0336	0.1261	0.5882	0.2269	0.0084	0.0000	0.0000	0.0000	0.0000	0.0000	0.1092	0.6891	0.2017	0.0000	0.0000	
13	0.0000	0.0000	0.0099	0.0099	0.0693	0.4554	0.4554	0.0000	0.0000	0.0000	0.0000	0.0000	0.0093	0.0792	0.5644	0.3366	0.0099	0.0000	
14	0.0000	0.0000	0.0106	0.0000	0.0638	0.5426	0.3830	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0517	0.5517	0.3793	0.0172	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0290	0.4783	0.4638	0.0290	0.0000	0.0000	0.0000	0.0000	0.0000	0.0250	0.6000	0.3750	0.0000	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.7000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.3556	0.0222	0.0222	
17	0.0000	0.0000	0.0000	0.0000	0.0172	0.2931	0.5862	0.1034	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.4750	0.2000	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0192	0.3077	0.6346	0.0385	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.4750	0.2000	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.2955	0.5000	0.1818	0.0227	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.6389	0.1389	0.0278	
20	0.0000	0.0000	0.0000	0.0000	0.0169	0.1695	0.5085	0.2542	0.0508	0.0000	0.0000	0.0000	0.0000	0.0000	0.1071	0.6429	0.1429	0.1071	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2045	0.5000	0.2500	0.0455	0.0000	0.0000	0.0000	0.0000	0.0000	0.0455	0.6818	0.1364	0.1364	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0909	0.4242	0.3939	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.1786	0.0714	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.2222	0.3333	0.2222	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2222	0.4444	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000	0.2000	0.6000	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.385	0.2308	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.1429	0.1429	0.5714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000	0.8000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

4100 EM										3800 EN									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0039	0.0174	0.2161	0.6043	0.1577	0.0006	0.0000	0.0000	0.0000	0.0070	0.0303	0.2242	0.6959	0.0419	0.0008	0.0000	0.0000	0.0000	
4	0.0028	0.0102	0.1061	0.5053	0.3753	0.0004	0.0000	0.0000	0.0000	0.0103	0.0154	0.0969	0.6295	0.2470	0.0009	0.0000	0.0000	0.0000	
5	0.0009	0.0059	0.0286	0.3338	0.6135	0.0168	0.0005	0.0000	0.0000	0.0036	0.0036	0.0392	0.4097	0.5333	0.0107	0.0000	0.0000	0.0000	
6	0.0015	0.0031	0.0214	0.2369	0.5435	0.1937	0.0000	0.0000	0.0000	0.0029	0.0044	0.0250	0.1956	0.7147	0.0559	0.0015	0.0000	0.0000	
7	0.0008	0.0000	0.0212	0.1715	0.4178	0.3887	0.0000	0.0000	0.0000	0.0047	0.0062	0.0156	0.1120	0.6563	0.2022	0.0031	0.0000	0.0000	
8	0.0016	0.0024	0.0162	0.1170	0.3193	0.5418	0.0016	0.0000	0.0000	0.0000	0.0000	0.0063	0.0877	0.4843	0.4217	0.0000	0.0000	0.0000	
9	0.0000	0.0012	0.0069	0.0822	0.2257	0.6505	0.0336	0.0000	0.0000	0.0000	0.0000	0.0121	0.0604	0.2779	0.6465	0.0000	0.0000	0.0000	
10	0.0017	0.0000	0.0034	0.0310	0.1824	0.6627	0.1188	0.0000	0.0000	0.0000	0.0000	0.0040	0.0356	0.2134	0.7075	0.0395	0.0000	0.0000	
11	0.0016	0.0000	0.0016	0.0231	0.0906	0.6442	0.2389	0.0000	0.0000	0.0000	0.0000	0.0000	0.0130	0.1000	0.6783	0.2043	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0042	0.0189	0.0692	0.4990	0.4025	0.0000	0.0000	0.0000	0.0000	0.0172	0.0920	0.6262	0.2804	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0072	0.0600	0.3741	0.5036	0.0552	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.5375	0.4000	0.0125	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0348	0.3513	0.5316	0.0823	0.0000	0.0000	0.0000	0.0000	0.0000	0.0820	0.3443	0.5328	0.0410	0.0000	
15	0.0000	0.0000	0.0000	0.0070	0.0279	0.2927	0.5366	0.1324	0.0035	0.0000	0.0000	0.0000	0.0000	0.0405	0.3986	0.5135	0.0405	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0183	0.2271	0.4908	0.2491	0.0147	0.0000	0.0000	0.0000	0.0000	0.0201	0.3624	0.4899	0.1275	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0105	0.2613	0.4390	0.2683	0.0209	0.0000	0.0000	0.0000	0.0000	0.0238	0.4048	0.4603	0.0952	0.0159	
18	0.0000	0.0000	0.0000	0.0000	0.0128	0.2426	0.4468	0.2809	0.0170	0.0000	0.0000	0.0000	0.0000	0.0213	0.3475	0.4681	0.0933	0.0638	
19	0.0000	0.0000	0.0000	0.0000	0.0115	0.2754	0.3237	0.3188	0.0628	0.0000	0.0000	0.0000	0.0000	0.0098	0.3235	0.4902	0.1569	0.0196	
20	0.0000	0.0000	0.0000	0.0000	0.0066	0.1645	0.2829	0.4671	0.0789	0.0000	0.0000	0.0000	0.0000	0.0137	0.1233	0.4932	0.2466	0.1233	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0928	0.2784	0.4124	0.2165	0.0000	0.0000	0.0000	0.0000	0.0000	0.0732	0.5366	0.2195	0.1707	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2468	0.3896	0.3636	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.4286	0.3095	0.1190	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.5000	0.3125	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.4074	0.1852	0.3704	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.2174	0.3696	0.3696	0.0000	0.0000	0.0000	0.0000	0.0000	0.0400	0.4400	0.0400	0.4800	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2051	0.3333	0.4615	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5556	0.2222	0.2222	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1905	0.3333	0.4762	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3889	0.4444	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1176	0.2941	0.5882	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.2308	0.6154	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.8750	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

		5380 EQ										1000 ET									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9			
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000			
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000			
3	0.0104	0.0096	0.4563	0.4792	0.0146	0.0000	0.0000	0.0000	0.0000	0.0004	0.0045	0.0530	0.7332	0.2086	0.0004	0.0000	0.0000	0.0000			
4	0.0162	0.0216	0.2216	0.5378	0.1946	0.0081	0.0000	0.0000	0.0000	0.0004	0.0020	0.0507	0.5091	0.4375	0.0004	0.0000	0.0000	0.0000			
5	0.0021	0.0231	0.1111	0.4738	0.3899	0.0000	0.0000	0.0000	0.0000	0.0008	0.0033	0.0388	0.2629	0.6878	0.0063	0.0000	0.0000	0.0000			
6	0.0037	0.0037	0.0586	0.3370	0.5861	0.0110	0.0000	0.0000	0.0000	0.0015	0.0010	0.0210	0.1392	0.6675	0.1699	0.0000	0.0000	0.0000			
7	0.0000	0.0000	0.0250	0.1900	0.6850	0.1000	0.0000	0.0000	0.0000	0.0008	0.0000	0.0111	0.0903	0.4192	0.4778	0.0008	0.0000	0.0000			
8	0.0000	0.0056	0.0169	0.0791	0.6554	0.2429	0.0000	0.0000	0.0000	0.0010	0.0000	0.0062	0.0450	0.2698	0.6790	0.0000	0.0000	0.0000			
9	0.0000	0.0000	0.0078	0.0703	0.4922	0.4219	0.0000	0.0078	0.0000	0.0000	0.0000	0.0039	0.0324	0.2311	0.7148	0.0167	0.0000	0.0000			
10	0.0000	0.0000	0.0165	0.0661	0.3388	0.5785	0.0000	0.0000	0.0000	0.0000	0.0000	0.0058	0.0203	0.1250	0.7267	0.1221	0.0000	0.0000			
11	0.0000	0.0000	0.0000	0.0612	0.2857	0.6327	0.0204	0.0000	0.0000	0.0000	0.0000	0.0000	0.0139	0.0780	0.6586	0.2496	0.0000	0.0000			
12	0.0000	0.0000	0.0000	0.0172	0.2069	0.7069	0.0690	0.0000	0.0000	0.0000	0.0000	0.0000	0.0112	0.0828	0.6219	0.2841	0.0000	0.0000			
13	0.0000	0.0000	0.0000	0.0000	0.1212	0.7576	0.1212	0.0000	0.0000	0.0000	0.0000	0.0000	0.0103	0.0412	0.6151	0.3333	0.0000	0.0000			
14	0.0000	0.0000	0.0000	0.0000	0.0725	0.6812	0.2319	0.0000	0.0145	0.0000	0.0000	0.0000	0.0000	0.0388	0.4341	0.5039	0.0233	0.0000			
15	0.0000	0.0000	0.0000	0.0137	0.0959	0.6712	0.2192	0.0000	0.0000	0.0000	0.0000	0.0038	0.0000	0.0303	0.4129	0.5265	0.0265	0.0000			
16	0.0000	0.0000	0.0000	0.0000	0.0667	0.7778	0.1333	0.0222	0.0000	0.0000	0.0000	0.0000	0.0035	0.0104	0.3472	0.5625	0.0764	0.0000			
17	0.0000	0.0000	0.0000	0.0000	0.0328	0.6230	0.3115	0.0328	0.0000	0.0000	0.0000	0.0000	0.0000	0.0072	0.2899	0.5870	0.1159	0.0000			
18	0.0000	0.0000	0.0000	0.0000	0.0588	0.5588	0.3529	0.0294	0.0000	0.0000	0.0000	0.0000	0.0000	0.0108	0.1720	0.5341	0.2509	0.0323			
19	0.0000	0.0000	0.0000	0.0000	0.0159	0.5873	0.2698	0.0952	0.0317	0.0000	0.0000	0.0000	0.0000	0.0040	0.1694	0.4960	0.2621	0.0685			
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.5547	0.3359	0.0859	0.0234	0.0000	0.0000	0.0000	0.0000	0.0000	0.1532	0.4511	0.3319	0.0638			
21	0.0000	0.0000	0.0000	0.0000	0.0328	0.3607	0.4098	0.1311	0.0656	0.0000	0.0000	0.0000	0.0000	0.0061	0.0727	0.4364	0.3697	0.1152			
22	0.0263	0.0000	0.0000	0.0000	0.0263	0.4474	0.3421	0.1579	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0096	0.3846	0.3654	0.2404			
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.2353	0.3824	0.2647	0.1176	0.0000	0.0000	0.0000	0.0000	0.0000	0.0274	0.2740	0.3425	0.3562			
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2143	0.1429	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0333	0.3333	0.3167	0.3167			
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7000	0.2000	0.1000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0278	0.1389	0.3056	0.5278			
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.0000	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1212	0.2121	0.6667			
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.2500	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0455	0.0909	0.3182	0.5455			
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.3000	0.6000			
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7500	0.0000	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3750	0.0000	0.6250			
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500			
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.2500			

TABLE B-6 (Continued)

0350 EW											5000 FN									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9		PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000		0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000		0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0018	0.0129	0.4118	0.4559	0.1176	0.0000	0.0000	0.0000	0.0000		0.0442	0.1405	0.8152	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
4	0.0022	0.0157	0.1816	0.3475	0.4529	0.0000	0.0000	0.0000	0.0000		0.0465	0.1214	0.8320	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
5	0.0035	0.0000	0.0490	0.1678	0.7797	0.0000	0.0000	0.0000	0.0000		0.0600	0.0467	0.8933	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0160	0.0840	0.6600	0.2400	0.0000	0.0000	0.0000		0.2308	0.1026	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0070	0.0775	0.3310	0.5845	0.0000	0.0000	0.0000		0.2500	0.1667	0.5833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0088	0.0531	0.2124	0.7257	0.0000	0.0000	0.0000		0.2500	0.3333	0.4167	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
9	0.0000	0.0000	0.0000	0.0101	0.2222	0.7576	0.0101	0.0000	0.0000		0.4000	0.0000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0370	0.2099	0.4815	0.0716	0.0000	0.0000		0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0189	0.0755	0.4717	0.4340	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0526	0.1579	0.3684	0.4211	0.0000	0.0000		0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
13	0.0000	0.0000	0.0217	0.0000	0.0870	0.3696	0.5000	0.0217	0.0000		0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0263	0.4737	0.4737	0.0263	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.8125	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0313	0.1875	0.7500	0.0313	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0323	0.2903	0.5806	0.0968	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.6154	0.2308	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.6000	0.1600	0.0400		0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1071	0.4643	0.3929	0.0357		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0800	0.5200	0.1200	0.2800		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0476	0.4286	0.3810	0.1429		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3529	0.3529	0.2941		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4615	0.2308	0.3077		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.3333		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.4000	0.2000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000		0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

0800 FT										0600 GM									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0013	0.0059	0.0703	0.6129	0.3097	0.0000	0.0000	0.0000	0.0000	0.0042	0.0209	0.2706	0.6301	0.0742	0.0000	0.0000	0.0000	0.0000	
4	0.0000	0.0038	0.0475	0.2753	0.6733	0.0000	0.0000	0.0000	0.0000	0.0022	0.0067	0.0762	0.5045	0.4081	0.0022	0.0000	0.0000	0.0000	
5	0.0000	0.0009	0.0263	0.1599	0.8074	0.0054	0.0000	0.0000	0.0000	0.0048	0.0000	0.0290	0.2738	0.6892	0.0032	0.0000	0.0000	0.0000	
6	0.0010	0.0019	0.0124	0.0975	0.6692	0.2170	0.0000	0.0000	0.0010	0.0019	0.0057	0.0133	0.0738	0.7738	0.0304	0.0019	0.0000	0.0000	
7	0.0000	0.0017	0.0050	0.0810	0.5322	0.3802	0.0000	0.0000	0.0000	0.0021	0.0042	0.0149	0.0870	0.6645	0.2251	0.0000	0.0021	0.0000	
8	0.0014	0.0014	0.0014	0.0291	0.4424	0.5243	0.0000	0.0000	0.0000	0.0022	0.0022	0.0131	0.0568	0.4258	0.5000	0.0000	0.0000	0.0000	
9	0.0022	0.0000	0.0043	0.0282	0.2733	0.6486	0.0434	0.0000	0.0000	0.0000	0.0000	0.0000	0.0214	0.2929	0.6810	0.0048	0.0000	0.0000	
10	0.0000	0.0022	0.0000	0.0112	0.2220	0.6300	0.1345	0.0000	0.0000	0.0000	0.0000	0.0029	0.0147	0.2006	0.7139	0.0678	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0075	0.1103	0.5639	0.3183	0.0000	0.0000	0.0035	0.0000	0.0070	0.0175	0.1404	0.6281	0.2035	0.0000	0.0000	
12	0.0027	0.0000	0.0000	0.0053	0.1277	0.5426	0.3218	0.0000	0.0000	0.0047	0.0000	0.0000	0.0094	0.0896	0.6792	0.2170	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0087	0.0652	0.4261	0.4913	0.0087	0.0000	0.0000	0.0000	0.0000	0.0163	0.1179	0.5366	0.3293	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0051	0.0513	0.3795	0.5333	0.0308	0.0000	0.0000	0.0000	0.0000	0.0112	0.0485	0.5299	0.4067	0.0037	0.0000	
15	0.0000	0.0000	0.0000	0.0048	0.0337	0.3510	0.5481	0.0625	0.0000	0.0000	0.0000	0.0043	0.0000	0.0433	0.4502	0.4892	0.0130	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0279	0.3184	0.5531	0.0950	0.0056	0.0000	0.0000	0.0000	0.0000	0.0461	0.3355	0.5789	0.0395	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0164	0.1694	0.5191	0.2896	0.0055	0.0000	0.0000	0.0000	0.0000	0.0455	0.2597	0.6558	0.0325	0.0065	
18	0.0000	0.0000	0.0000	0.0000	0.0059	0.1588	0.4882	0.3235	0.0235	0.0000	0.0000	0.0000	0.0000	0.0467	0.3178	0.4860	0.1308	0.0187	
19	0.0000	0.0000	0.0000	0.0000	0.0067	0.1275	0.3624	0.4497	0.0537	0.0000	0.0000	0.0000	0.0000	0.0088	0.2920	0.5929	0.0708	0.0354	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1078	0.3431	0.4020	0.1471	0.0000	0.0000	0.0000	0.0101	0.0404	0.2929	0.5354	0.1212	0.0000	
21	0.0000	0.0000	0.0000	0.0114	0.0000	0.0568	0.3977	0.3977	0.1364	0.0000	0.0000	0.0000	0.0096	0.0096	0.1923	0.5865	0.1827	0.0192	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0641	0.2179	0.3846	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0556	0.6667	0.2037	0.0741	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0127	0.2025	0.3797	0.4051	0.0000	0.0000	0.0000	0.0000	0.0000	0.1220	0.5366	0.2195	0.1220	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213	0.1702	0.2553	0.5532	0.0000	0.0000	0.0000	0.0000	0.0000	0.0488	0.4390	0.3171	0.1951	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.5089	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213	0.5106	0.2979	0.1702	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1579	0.2632	0.5789	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3158	0.5789	0.1053	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.5000	0.3750	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.2778	0.2778	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.3333	0.4167	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2667	0.3333	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.1429	0.7143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.3125	0.1250	0.5000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2857	0.5714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.5000	0.1000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2222	0.2222	0.5556	



TABLE B-6 (Continued)

4400 GS										8000 L.A									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0068	0.0113	0.1769	0.6122	0.1927	0.0000	0.0000	0.0000	0.0000	0.0052	0.0241	0.6411	0.3141	0.0143	0.0009	0.0000	0.0003	0.0000	
4	0.0048	0.0072	0.1106	0.4231	0.4543	0.0000	0.0000	0.0000	0.0000	0.0043	0.0148	0.3428	0.5459	0.0914	0.0007	0.0000	0.0000	0.0000	
5	0.0000	0.0037	0.0319	0.2259	0.7111	0.0074	0.0000	0.0000	0.0000	0.0037	0.0050	0.1156	0.5539	0.3172	0.0046	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0160	0.0957	0.6649	0.2234	0.0000	0.0000	0.0000	0.0023	0.0037	0.0496	0.4286	0.5032	0.1116	0.0005	0.0005	0.0000	
7	0.0000	0.0000	0.0388	0.1008	0.4109	0.4496	0.0000	0.0000	0.0000	0.0013	0.0019	0.0302	0.2770	0.6228	0.0662	0.0006	0.0000	0.0000	
8	0.0000	0.0000	0.0089	0.0714	0.3214	0.5982	0.0000	0.0000	0.0000	0.0008	0.0008	0.0280	0.1454	0.6550	0.1693	0.0008	0.0000	0.0000	
9	0.0000	0.0000	0.0000	0.0311	0.2671	0.6335	0.0683	0.0000	0.0000	0.0045	0.0022	0.0089	0.0737	0.5547	0.3482	0.0078	0.0000	0.0000	
10	0.0000	0.0000	0.0183	0.0275	0.1468	0.6606	0.1468	0.0000	0.0000	0.0000	0.0027	0.0082	0.0721	0.3959	0.4939	0.0272	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0123	0.0988	0.5679	0.3086	0.0000	0.0123	0.0015	0.0000	0.0089	0.0355	0.2352	0.6302	0.0888	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.0615	0.5231	0.4154	0.0000	0.0000	0.0017	0.0017	0.0017	0.0310	0.2397	0.6052	0.1190	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0149	0.0448	0.3582	0.5821	0.0000	0.0000	0.0000	0.0000	0.0051	0.0306	0.1446	0.5986	0.2211	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0192	0.0385	0.2500	0.6538	0.0385	0.0000	0.0000	0.0000	0.0022	0.0246	0.1119	0.4989	0.3624	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0625	0.2344	0.6250	0.0781	0.0000	0.0000	0.0000	0.0039	0.0059	0.0391	0.4629	0.4551	0.0332	0.0021	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.1579	0.7105	0.1053	0.0263	0.0000	0.0000	0.0000	0.0000	0.0036	0.0308	0.4004	0.4982	0.0670	
17	0.0000	0.0000	0.0000	0.0294	0.0294	0.0588	0.5000	0.3235	0.0588	0.0000	0.0000	0.0000	0.0025	0.0098	0.3145	0.5405	0.1229	0.0098	
18	0.0000	0.0000	0.0000	0.0000	0.0200	0.1000	0.4200	0.3800	0.0800	0.0000	0.0000	0.0000	0.0041	0.0082	0.2510	0.5556	0.1811	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0968	0.5484	0.3226	0.0323	0.0000	0.0000	0.0000	0.0000	0.0000	0.0283	0.2105	0.5506	0.1781	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1786	0.4643	0.2857	0.0714	0.0000	0.0000	0.0000	0.0000	0.0076	0.2214	0.4504	0.2366	0.0840	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4167	0.3333	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.1038	0.4623	0.3208	0.1132	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0941	0.3882	0.4000	0.1176	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0392	0.4314	0.3725	0.1569	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.0116	0.0000	0.0000	0.0000	0.0233	0.3953	0.3140	0.2558	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0130	0.3377	0.2727	0.3766	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2549	0.4902	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1613	0.3226	0.5161	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3448	0.3448	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2400	0.5200	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1176	0.7647	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

4303 HT

2300 IS

LOS	PAY1	PAY2	PAY3	PAY4	PAY5	PAY6	PAY7	PAY8	PAY9	PAY1	PAY2	PAY3	PAY4	PAY5	PAY6	PAY7	PAY8	PAY9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0084	0.0290	0.2297	0.6389	0.0929	0.0000	0.0011	0.0000	0.0000	0.0000	0.0056	0.1404	0.6404	0.2079	0.0056	0.0000	0.0000	0.0000
4	0.0020	0.0138	0.0813	0.5197	0.3814	0.0020	0.0000	0.0000	0.0000	0.0000	0.0068	0.0544	0.2925	0.6395	0.0068	0.0000	0.0000	0.0000
5	0.0044	0.0033	0.0316	0.2832	0.6514	0.0261	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213	0.0638	0.8723	0.0426	0.0000	0.0000	0.0000
6	0.0034	0.0023	0.0159	0.1506	0.7520	0.0736	0.0011	0.0011	0.0000	0.0000	0.0000	0.0000	0.0526	0.8289	0.1184	0.0000	0.0000	0.0000
7	0.0012	0.0012	0.0170	0.0887	0.6561	0.2345	0.0012	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.5435	0.4022	0.0109	0.0000	0.0000
8	0.0015	0.0015	0.0161	0.0585	0.4737	0.4488	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0225	0.1236	0.8202	0.0225	0.0000	0.0000
9	0.0000	0.0020	0.0059	0.0475	0.2733	0.6554	0.0139	0.0000	0.0020	0.0000	0.0000	0.0000	0.0225	0.1236	0.8202	0.0225	0.0000	0.0000
10	0.0000	0.0054	0.0081	0.0323	0.2022	0.7116	0.0404	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.0526	0.7895	0.1053	0.3000	0.0000
11	0.0000	0.0000	0.0065	0.0163	0.1889	0.6156	0.1661	0.0065	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.7813	0.1563	0.0000	0.0000
12	0.0000	0.0000	0.0030	0.0152	0.0973	0.6626	0.2188	0.0030	0.0000	0.0000	0.0000	0.0000	0.0000	0.1143	0.6040	0.2857	0.0000	0.0000
13	0.0000	0.0000	0.0031	0.0061	0.0767	0.5828	0.3282	0.0031	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7714	0.2286	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0185	0.0463	0.4969	0.4259	0.0123	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4333	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0036	0.0287	0.4409	0.4982	0.0251	0.0036	0.0000	0.0000	0.0000	0.0000	0.0333	0.5333	0.4333	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0149	0.3119	0.5990	0.0743	0.0000	0.0000	0.0000	0.0000	0.0000	0.0345	0.4828	0.4828	0.0000	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0284	0.2796	0.5545	0.1327	0.0047	0.0000	0.0000	0.0000	0.0000	0.0000	0.6071	0.3214	0.0714	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0387	0.2774	0.5032	0.1548	0.0258	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.5556	0.1111	0.0000
19	0.0000	0.0000	0.0000	0.0065	0.0131	0.2092	0.5817	0.1634	0.0261	0.0000	0.0000	0.0000	0.0000	0.0526	0.2632	0.4211	0.2105	0.0526
20	0.0000	0.0000	0.0000	0.0000	0.0139	0.2222	0.4167	0.2778	0.0694	0.0000	0.0000	0.0000	0.0000	0.0000	0.3077	0.3846	0.3077	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0086	0.1034	0.5431	0.2845	0.0603	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.2857	0.2857	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.4630	0.3148	0.0556	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.1667	0.1667
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0566	0.3962	0.3208	0.2264	0.0000	0.0000	0.0000	0.0000	0.0000	0.3750	0.3750	0.1250	0.1250
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0476	0.3333	0.2381	0.3810	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.2857	0.4286
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0357	0.2500	0.2143	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2353	0.2941	0.4706	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.2857	0.4286
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.1250	0.6875	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2941	0.7059	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.1667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.1818	0.6364	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



TABLE B-6 (Continued)

2600 JO

3100 LI

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0000	0.0106	0.2553	0.5745	0.1489	0.0106	0.0000	0.0000	0.0000	0.0333	0.0333	0.1667	0.6333	0.1333	0.0000	0.0000	0.0000	0.0000
4	0.0133	0.0000	0.0667	0.5867	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8250	0.1750	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0513	0.3846	0.5256	0.0256	0.0128	0.0000	0.0000	0.0000	0.0204	0.0612	0.5714	0.3469	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.1833	0.7500	0.0500	0.0167	0.0000	0.0000	0.0000	0.0000	0.0385	0.4038	0.5577	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.0968	0.6774	0.2258	0.0000	0.0000	0.0000	0.0000	0.0000	0.0217	0.2391	0.6957	0.0435	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0517	0.1034	0.4483	0.3793	0.0172	0.0000	0.0000	0.0000	0.0000	0.0345	0.1724	0.5862	0.2069	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0192	0.0577	0.4231	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.5313	0.4063	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0833	0.3889	0.5278	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.2800	0.0000	0.1600	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.0000	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.1111	0.5926	0.2593	0.0000	0.0370	0.0000	0.0000	0.0000	0.1053	0.3684	0.4211	0.1053	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.1053	0.5263	0.3421	0.0263	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.7500	0.1500	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0667	0.5000	0.3667	0.0667	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.5000	0.3000	0.1000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.5909	0.4091	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7059	0.1176	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.5357	0.0357	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7500	0.1250	0.1250	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.3077	0.4615	0.2308	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6923	0.3077	0.0000	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0588	0.1765	0.5294	0.1765	0.0588	0.0000	0.0000	0.0000	0.0000	0.0714	0.4286	0.3571	0.1429	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.2353	0.2941	0.4118	0.0588	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.6250	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.8750	0.0000	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.3636	0.4545	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.6000	0.0000	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.3333	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.4444	0.3333	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.4286	0.1429	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.2500	0.2500
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-6 (Continued)

1750 LN

0150 MA

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.2000	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.2000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	0.9474	0.0526	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6579	0.3421	0.3333	0.0222	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.5769	0.3846	0.0385	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6444	0.3555	0.6271	0.0169	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.5094	0.4906	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.3448	0.6552	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3559	0.6441	0.6271	0.0169	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.7143	0.6825	0.0317	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.1364	0.7727	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1264	0.8736	0.8046	0.0690	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.0435	0.6957	0.2609	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0270	0.9730	0.8108	0.1622	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.0455	0.6818	0.2727	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0667	0.9333	0.7867	0.1467	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0455	0.5000	0.4545	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0265	0.9735	0.6460	0.3186	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.5263	0.4737	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0174	0.9826	0.6174	0.3478	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.4500	0.5500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5246	0.4344	0.4040	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0667	0.4000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0175	0.4035	0.5263	0.0526	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.2083	0.6667	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0097	0.4466	0.4466	0.0777	0.0194
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.7000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4132	0.4713	0.1034	0.0115
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.4444	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3412	0.4824	0.1294	0.0471
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.4545	0.2727	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.2424	0.4646	0.2222	0.0707
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.5455	0.0909	0.0909	0.0000	0.0000	0.0000	0.0000	0.0000	0.1277	0.4043	0.3191	0.1489
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1290	0.3548	0.3548	0.1613
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.1786	0.3571	0.2143	0.2500
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.3333	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.3000	0.4500	0.1500
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.2000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.2500	0.2500
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.5714
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5714	0.2857
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.8333
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	0.8000

TABLE B-6 (Continued)

4700 ML

3700 MM

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0084	0.0262	0.2301	0.5609	0.1741	0.0003	0.0000	0.0000	0.0000
4	0.0000	0.0345	0.4828	0.0345	0.4828	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.2105	0.7895	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0667	0.8667	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.1429	0.6429	0.2143	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.2000	0.6000	0.2000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.3333	0.3333	0.6667	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.1111	0.4444	0.4444	0.0000	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.2500	0.2500	0.2500	0.5000	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	0.0000	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.5714	0.2857	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	0.0000	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500	0.0000	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.0000	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

TABLE B-6 (Continued)

0900 MN										3900 MR									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0000	0.0339	0.2034	0.7627	0.0000	0.0000	0.0000	0.0000	0.0000	0.0062	0.0227	0.2045	0.6674	0.0092	0.0000	0.0000	0.0000	0.0000	
4	0.0000	0.0192	0.0962	0.8077	0.0769	0.0000	0.0000	0.0000	0.0000	0.0067	0.0044	0.0778	0.4178	0.4933	0.0000	0.0000	0.0000	0.0000	
5	0.0000	0.0000	0.0976	0.4390	0.4390	0.0244	0.0000	0.0000	0.0000	0.0041	0.0083	0.0373	0.2324	0.7012	0.0166	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0606	0.3333	0.6061	0.0000	0.0000	0.0000	0.0000	0.0047	0.0047	0.0188	0.1784	0.7700	0.0235	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0000	0.4667	0.5000	0.0333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0829	0.7358	0.1762	0.0052	0.0000	0.0000	
8	0.0000	0.0000	0.0000	0.1163	0.7442	0.1395	0.6000	0.0000	0.0000	0.0000	0.0000	0.0075	0.0526	0.5714	0.3684	0.0000	0.0000	0.0000	
9	0.0000	0.0000	0.0000	0.0606	0.5152	0.4242	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0602	0.5181	0.4096	0.0120	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0833	0.3333	0.5833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0115	0.0115	0.3333	0.6322	0.0115	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0435	0.1304	0.6087	0.2174	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1546	0.7835	0.0619	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.4000	0.5000	0.0562	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100	0.0900	0.8500	0.0500	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.5000	0.3571	0.0000	0.0000	0.0000	0.0000	0.0000	0.1236	0.6854	0.1910	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0878	0.6522	0.2609	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0882	0.7206	0.1912	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0789	0.5263	0.3684	0.0263	0.0000	0.0000	0.0000	0.0000	0.0000	0.0241	0.6145	0.3373	0.0241	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0769	0.6154	0.3077	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0227	0.6136	0.3182	0.0455	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0714	0.5000	0.2143	0.2143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.5000	0.3043	0.1522	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.1111	0.4444	0.3333	0.1111	0.0000	0.0000	0.0000	0.0000	0.0000	0.0132	0.5395	0.3816	0.0658	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0513	0.4615	0.3333	0.1026	0.0513	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.1667	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.4490	0.3878	0.0612	0.1020	
21	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.3333	0.4815	0.1481	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500	0.2857	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.3333	0.2222	0.3333	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	
25	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.2000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	0.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	

TABLE B-6 (Continued)

2200 MS										3300 MU									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0150	0.0590	0.3632	0.5466	0.0150	0.0012	0.0000	0.0000	0.0000	0.0000	0.0000	0.1525	0.6780	0.1695	0.0000	0.0000	0.0000	0.0000	
4	0.0126	0.0319	0.1750	0.6951	0.0830	0.0024	0.0000	0.0000	0.0000	0.0000	0.0114	0.0568	0.5795	0.2386	0.1136	0.0000	0.0000	0.0000	
5	0.0070	0.0087	0.0814	0.6115	0.2905	0.0009	0.0000	0.0000	0.0000	0.0000	0.0000	0.0476	0.4603	0.2540	0.2381	0.0000	0.0000	0.0000	
6	0.0053	0.0036	0.0454	0.4898	0.4541	0.0018	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.4400	0.1200	0.0400	0.0000	0.0000	
7	0.0034	0.0034	0.0263	0.3707	0.5892	0.0069	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.3500	0.0500	0.0000	0.0000	
8	0.0037	0.0037	0.0337	0.2643	0.6571	0.0362	0.0012	0.0000	0.0000	0.0000	0.0000	0.0000	0.1842	0.5263	0.2105	0.0789	0.0000	0.0000	
9	0.0013	0.0025	0.0177	0.1962	0.6633	0.1177	0.0013	0.0000	0.0000	0.0000	0.0000	0.0217	0.1957	0.5870	0.1739	0.0217	0.0000	0.0000	
10	0.0000	0.0049	0.0211	0.1542	0.6201	0.1981	0.0016	0.0000	0.0000	0.0000	0.0000	0.0000	0.1081	0.4865	0.3514	0.0541	0.0000	0.0000	
11	0.0021	0.0000	0.0063	0.0814	0.5804	0.3215	0.0084	0.0000	0.0000	0.0000	0.0000	0.0625	0.1250	0.2500	0.5625	0.0000	0.0000	0.0000	
12	0.0000	0.0026	0.0104	0.0961	0.5377	0.3065	0.0468	0.0000	0.0000	0.0000	0.0000	0.0000	0.0556	0.1667	0.5000	0.2222	0.0556	0.0000	
13	0.0029	0.0000	0.0029	0.0523	0.5000	0.3895	0.0494	0.0029	0.0000	0.0000	0.0000	0.0000	0.0263	0.1842	0.6316	0.1316	0.0263	0.0000	
14	0.0063	0.0000	0.0000	0.0476	0.4571	0.4127	0.0730	0.0032	0.0000	0.0000	0.0000	0.0000	0.0000	0.3056	0.5278	0.1389	0.0278	0.0000	
15	0.0000	0.0000	0.0000	0.0424	0.3517	0.4661	0.1398	0.0000	0.0000	0.0000	0.0000	0.0278	0.0556	0.1111	0.6389	0.1667	0.0000	0.0000	
16	0.0000	0.0000	0.0000	0.0134	0.2198	0.6032	0.1635	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1724	0.4828	0.3103	0.0345	0.0000	
17	0.0000	0.0021	0.0000	0.0084	0.1282	0.6345	0.2017	0.0210	0.0042	0.0000	0.0000	0.0000	0.0435	0.1304	0.3043	0.3043	0.1739	0.0435	
18	0.0011	0.0000	0.0000	0.0043	0.0944	0.6631	0.2092	0.0247	0.0032	0.0000	0.0000	0.0000	0.0000	0.0741	0.3704	0.4444	0.0741	0.0370	
19	0.0000	0.0000	0.0000	0.0036	0.0696	0.6125	0.2518	0.0571	0.0054	0.0000	0.0000	0.0000	0.0000	0.3043	0.3043	0.1739	0.2174	0.0000	
20	0.0000	0.0000	0.0000	0.0016	0.0750	0.5703	0.2594	0.0750	0.0188	0.0000	0.0000	0.0000	0.0000	0.1739	0.3043	0.3478	0.1304	0.0435	
21	0.0000	0.0000	0.0015	0.0000	0.0437	0.5053	0.3484	0.0890	0.0121	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.6000	0.0000	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0175	0.3202	0.4649	0.1623	0.0351	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.3750	0.1250	0.0000	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.1857	0.4143	0.2786	0.1214	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.1667	0.3333	0.1667	
24	0.0000	0.0000	0.0000	0.0000	0.0130	0.1299	0.3506	0.3247	0.1818	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.5556	0.3333	0.0000	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0870	0.4261	0.2870	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0652	0.4130	0.2609	0.2609	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	
27	0.0000	0.0000	0.0000	0.0000	0.0182	0.0364	0.3455	0.2909	0.3091	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.2500	0.4167	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.0769	0.4615	0.4359	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.1538	0.7692	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.2308	0.2308	0.4615	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.2500	0.5833	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

1400 NC

0300 OS

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0111	0.0350	0.3032	0.5988	0.0513	0.0006	0.0000	0.0000	0.0000
4	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0057	0.0150	0.0991	0.6009	0.2780	0.0007	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0048	0.0072	0.0442	0.3441	0.5884	0.0113	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0041	0.0055	0.0194	0.2019	0.7123	0.0567	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0020	0.0039	0.0275	0.1415	0.5658	0.2574	0.0020	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0000	0.0625	0.9375	0.0000	0.0000	0.0000	0.0020	0.0039	0.0275	0.1415	0.5658	0.2574	0.0020	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0000	0.0526	0.9211	0.0263	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0000	0.0135	0.8514	0.1216	0.0135	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.0123	0.8148	0.1728	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.2000	0.0000	0.0000	0.0035	0.0000	0.0035	0.0521	0.1215	0.6667	0.1528	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.6600	0.3300	0.0000	0.0000	0.0000	0.0000	0.0000	0.0100	0.1650	0.5950	0.2300	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0000	0.5804	0.3929	0.0268	0.0000	0.0045	0.0000	0.0000	0.0090	0.1126	0.5676	0.3063	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.4628	0.4876	0.0413	0.0083	0.0000	0.0000	0.0000	0.0000	0.0479	0.4452	0.4589	0.0479	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.4595	0.4595	0.0721	0.0090	0.0000	0.0000	0.0000	0.0000	0.0259	0.4224	0.4655	0.0862	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0070	0.4965	0.4336	0.0559	0.0070	0.0000	0.0000	0.0000	0.0000	0.0439	0.2193	0.4211	0.2807	0.0351
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.3163	0.4490	0.2143	0.0204	0.0000	0.0000	0.0000	0.0000	0.0316	0.2105	0.4105	0.3263	0.0211
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.3049	0.5244	0.1463	0.0244	0.0000	0.0000	0.0000	0.0147	0.0294	0.2353	0.3676	0.3529	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1711	0.5658	0.2105	0.0526	0.0000	0.0000	0.0000	0.0000	0.0351	0.1930	0.3509	0.3860	0.0351
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.1176	0.5059	0.2471	0.1294	0.0000	0.0000	0.0000	0.0000	0.0000	0.1186	0.3898	0.4576	0.0339
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.4808	0.2885	0.0769	0.0000	0.0000	0.0000	0.0000	0.0000	0.1489	0.2553	0.4255	0.1702
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0938	0.4063	0.3750	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0800	0.2000	0.4400	0.2800
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0351	0.2281	0.4386	0.2982	0.0000	0.0000	0.0000	0.0000	0.0000	0.1034	0.3103	0.4828	0.1834
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3409	0.4318	0.2273	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2813	0.4375	0.2813
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1563	0.3125	0.5313	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0952	0.5714	0.3333
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1579	0.4211	0.4211	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.3846	0.4615
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2667	0.3333	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0952	0.6667	0.2381
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2353	0.2941	0.4706	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.5000	0.2000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2857	0.5714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2222	0.4444	0.3333
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.2000	0.6000



TABLE B-6 (Continued)

0450 OT

2700 PC

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0063	0.0125	0.1875	0.7063	0.0875	0.0000	0.0000	0.0000	0.0000	0.0058	0.0175	0.1111	0.8363	0.0292	0.0000	0.0000	0.0000	0.0000
4	0.0000	0.0139	0.0741	0.5046	0.4074	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0341	0.7216	0.2443	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0109	0.2077	0.7814	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0404	0.4444	0.5152	0.0000	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0000	0.1047	0.8779	0.0174	0.0000	0.0000	0.0000	0.0000	0.0000	0.0208	0.4167	0.5625	0.0000	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0198	0.0891	0.4752	0.4158	0.0000	0.0000	0.0000	0.0000	0.0000	0.0455	0.2159	0.7045	0.0341	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0106	0.0638	0.5213	0.3936	0.0106	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.5469	0.2656	0.0000	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0635	0.2222	0.7143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.0769	0.6154	0.2821	0.0000	0.0000	0.0000
10	0.0000	0.0000	0.0000	0.0323	0.1935	0.7742	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.5476	0.0238	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0270	0.1216	0.6757	0.1757	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6250	0.0313	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.0488	0.8049	0.1463	0.0000	0.0000	0.0000	0.0000	0.0000	0.0556	0.2222	0.6389	0.0556	0.0278	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.0625	0.7292	0.2083	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0909	0.7727	0.1364	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0465	0.6744	0.2791	0.0000	0.0000	0.0000	0.0000	0.0000	0.0256	0.1026	0.7692	0.1026	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0392	0.6078	0.3333	0.0196	0.0000	0.0000	0.0000	0.0000	0.0000	0.1026	0.7179	0.1538	0.0256	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.5517	0.4483	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6842	0.3158	0.0000	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.2069	0.6897	0.1034	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.6000	0.3500	0.0000	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.4118	0.4118	0.1471	0.0294	0.0000	0.0000	0.0000	0.0000	0.0000	0.5263	0.3158	0.1053	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.1000	0.2667	0.4333	0.1667	0.0333	0.0000	0.0000	0.0000	0.0000	0.1250	0.2500	0.3750	0.2500	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.1786	0.6071	0.1786	0.0357	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3077	0.4615	0.0769
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.3529	0.3529	0.2941	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.1429	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5625	0.3125	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.2667	0.4667	0.2000	0.0667
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3750	0.3750	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.1429	0.1429	0.2857
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.4286	0.2857	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.3333	0.2222	0.3333
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4444	0.1111	0.4444
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.0000	0.8000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.2000	0.6000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000

TABLE B-6 (Continued)

7600 PH

1080 PI

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0061	0.0061	0.3497	0.6012	0.0368	0.0000	0.0000	0.0000	0.0000	0.0057	0.0172	0.1322	0.7586	0.0805	0.0057	0.0000	0.0000	0.0000
4	0.0000	0.0132	0.2781	0.5960	0.1126	0.0000	0.0000	0.0000	0.0000	0.0000	0.0069	0.0625	0.4722	0.4514	0.0069	0.0000	0.0000	0.0000
5	0.0000	0.0042	0.1186	0.5636	0.3136	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0215	0.2151	0.7527	0.0108	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0432	0.3669	0.5540	0.0360	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1690	0.7606	0.0704	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0211	0.2042	0.6761	0.0986	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.7917	0.1458	0.0000	0.0000	0.0000
8	0.0000	0.0000	0.0098	0.1176	0.6176	0.2549	0.0000	0.0000	0.0000	0.0000	0.0000	0.0233	0.0233	0.5116	0.4186	0.0233	0.0000	0.0000
9	0.0000	0.0101	0.0000	0.0909	0.5455	0.3535	0.0000	0.0000	0.0000	0.0000	0.0000	0.0556	0.0556	0.3056	0.5833	0.0233	0.0000	0.0000
10	0.0000	0.0000	0.0156	0.0938	0.4844	0.4063	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1923	0.7692	0.0385	0.0000	0.0000
11	0.0000	0.0000	0.0000	0.0294	0.2941	0.6471	0.0294	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2222	0.7407	0.0370	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0000	0.2174	0.6957	0.0870	0.0000	0.0000	0.0000	0.0000	0.0000	0.0294	0.0588	0.7941	0.1176	0.0000	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.1667	0.6667	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.7368	0.2105	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0286	0.1429	0.7429	0.0857	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.7368	0.2105	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.0400	0.8000	0.1200	0.0400	0.0000	0.0000	0.0000	0.0000	0.0000	0.0286	0.7143	0.2286	0.0286	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0278	0.7222	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0506	0.4242	0.4848	0.0303	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0233	0.6512	0.3023	0.0233	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.4583	0.0417	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.5405	0.3514	0.1081	0.0000	0.0000	0.0000	0.0000	0.0303	0.0000	0.2727	0.4848	0.1818	0.0303
19	0.0000	0.0000	0.0000	0.0000	0.0500	0.6250	0.3000	0.0000	0.0250	0.0000	0.0000	0.0000	0.0000	0.0000	0.5294	0.3529	0.1176	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.4375	0.5000	0.0625	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1875	0.5000	0.2500	0.0625
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.4615	0.3846	0.0000	0.1538	0.0000	0.0000	0.0000	0.0000	0.0769	0.2308	0.6154	0.0769	0.0000
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5833	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.5000	0.0000
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6250	0.2500	0.1250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	0.0000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.3000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.1667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.0000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.4286	0.1429	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.5000	0.2500
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.2500	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.2857	0.4286	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000



TABLE B-6 (Continued)

4500 PM										1800 PM									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0000	0.0400	0.0800	0.7200	0.1600	0.0000	0.0000	0.0000	0.0000	0.0035	0.0088	0.1684	0.7246	0.0895	0.0035	0.0018	0.0000	0.0000	
4	0.0000	0.0000	0.0526	0.3684	0.5789	0.0000	0.0000	0.0000	0.0000	0.0014	0.0070	0.1043	0.5744	0.3115	0.0014	0.0000	0.0000	0.0000	
5	0.0000	0.0000	0.0000	0.4444	0.5556	0.0000	0.0000	0.0000	0.0000	0.0000	0.0057	0.0355	0.3716	0.5730	0.0142	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0000	0.1111	0.8889	0.0000	0.0000	0.0000	0.0000	0.0000	0.0033	0.0280	0.2089	0.6974	0.0625	0.0000	0.0000	0.0000	
7	0.0000	0.0000	0.0000	0.1000	0.5000	0.4000	0.0000	0.0000	0.0000	0.0019	0.0000	0.0195	0.1618	0.6062	0.2105	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0000	0.1000	0.5000	0.3000	0.0000	0.0000	0.0000	0.0000	0.0019	0.0113	0.0679	0.4566	0.4604	0.0019	0.0000	0.0000	
9	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0021	0.0000	0.0021	0.0511	0.3340	0.6085	0.0021	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0196	0.2124	0.7484	0.0163	0.0033	0.0000	
11	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	0.0000	0.0000	0.0000	0.0035	0.0000	0.0174	0.0694	0.7674	0.1354	0.0069	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0066	0.0563	0.6854	0.2450	0.0066	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.2000	0.8000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0045	0.0407	0.5204	0.4072	0.0271	0.0000	
14	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0066	0.0596	0.3775	0.4834	0.0728	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.8333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0119	0.4107	0.5298	0.0357	0.0119	
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.4286	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0140	0.3357	0.4825	0.1469	0.0210	
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.8000	0.2000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0147	0.3235	0.4559	0.1618	0.0441	
18	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1810	0.5810	0.2000	0.0381	
19	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0101	0.1414	0.4949	0.2424	0.1111	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2317	0.5122	0.1829	0.0732	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1923	0.5000	0.1538	0.1538	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0566	0.5660	0.1887	0.1887	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0217	0.3261	0.2391	0.4130	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0541	0.3784	0.2432	0.3243	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2759	0.1724	0.5517	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0667	0.4667	0.4667	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1333	0.0667	0.8000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1538	0.2308	0.6154	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.1250	0.6250	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.0500	0.8500	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.8500	

TABLE B-6 (Continued)

7000 PR

0200 OM

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0041	0.0165	0.2593	0.6214	0.0988	0.0000	0.0000	0.0000	0.0000	0.0068	0.0237	0.2081	0.6633	0.0981	0.0000	0.0000	0.0000	0.0000
4	0.0044	0.0000	0.1289	0.4489	0.4178	0.0000	0.0000	0.0000	0.0000	0.0078	0.0078	0.0719	0.4766	0.4359	0.0000	0.0000	0.0000	0.0000
5	0.0000	0.0000	0.0449	0.2247	0.7247	0.0056	0.0000	0.0000	0.0000	0.0025	0.0025	0.0302	0.2217	0.7154	0.0277	0.0000	0.0000	0.0000
6	0.0000	0.0000	0.0160	0.1360	0.8480	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0204	0.1102	0.8000	0.0694	0.0000	0.0000	0.0000
7	0.0000	0.0000	0.0191	0.1338	0.8153	0.0318	0.0000	0.0000	0.0000	0.0000	0.0036	0.0072	0.1119	0.6137	0.2563	0.0072	0.0000	0.0000
8	0.0000	0.0000	0.0000	0.0377	0.7642	0.1981	0.0000	0.0000	0.0000	0.0115	0.0038	0.0038	0.0613	0.4751	0.4406	0.0038	0.0000	0.0000
9	0.0000	0.0000	0.0000	0.0603	0.5862	0.3534	0.0000	0.0000	0.0000	0.0000	0.0056	0.0111	0.0222	0.3222	0.6167	0.0222	0.0000	0.0000
10	0.0000	0.0000	0.0127	0.0380	0.5063	0.4304	0.0127	0.0000	0.0000	0.0000	0.0000	0.0078	0.0156	0.2344	0.5938	0.1406	0.0078	0.0000
11	0.0000	0.0000	0.0000	0.0000	0.3462	0.6410	0.0128	0.0000	0.0000	0.0000	0.0000	0.0000	0.0345	0.2069	0.5345	0.2241	0.0000	0.0000
12	0.0000	0.0000	0.0000	0.0139	0.2639	0.6806	0.0417	0.0000	0.0000	0.0000	0.0000	0.0000	0.0104	0.1354	0.5000	0.3438	0.0104	0.0000
13	0.0000	0.0000	0.0000	0.0000	0.2603	0.6986	0.0411	0.0000	0.0000	0.0000	0.0000	0.0000	0.0263	0.1140	0.3509	0.5088	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0000	0.0896	0.8209	0.0896	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.3913	0.5565	0.0087	0.0000
15	0.0000	0.0000	0.0000	0.0000	0.1587	0.7460	0.0952	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0531	0.2566	0.6372	0.0442	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0488	0.6585	0.2439	0.0488	0.0000	0.0000	0.0000	0.0000	0.0000	0.0519	0.2338	0.6623	0.0519	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0238	0.6429	0.2143	0.0952	0.0238	0.0000	0.0000	0.0000	0.0000	0.0253	0.1139	0.7215	0.1392	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0250	0.6000	0.2500	0.1000	0.0250	0.0000	0.0000	0.0000	0.0200	0.0000	0.1000	0.7000	0.1400	0.0400
19	0.0000	0.0000	0.0000	0.0000	0.0714	0.5714	0.2143	0.1429	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1290	0.5161	0.3548	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.4286	0.3929	0.1429	0.0357	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.6232	0.3188	0.0145
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.5714	0.2143	0.2143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0303	0.6061	0.2727	0.0909
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.6471	0.1765	0.1765	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0250	0.6000	0.3250	0.0500
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.5714	0.2857	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.5750	0.2750	0.1000
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3125	0.1875	0.1875	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4800	0.3600	0.1600
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.2500	0.4167	0.1667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0645	0.3871	0.3548	0.1935
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.2500	0.2500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4167	0.3333
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3750	0.3750	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.2500	0.5000
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.5000	0.1667
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.3333
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.0000	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2143	0.2857	0.5000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000

TABLE B-6 (Continued)

1500 RM										2500 RP									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0075	0.0303	0.3972	0.5179	0.0464	0.0008	0.0000	0.0000	0.0000	0.0000	0.0000	0.1318	0.7442	0.1240	0.0000	0.0000	0.0000	0.0000	
4	0.0045	0.0154	0.1140	0.6440	0.2221	0.0000	0.0000	0.0000	0.0000	0.0000	0.0084	0.0672	0.5210	0.4034	0.0000	0.0000	0.0000	0.0000	
5	0.0020	0.0052	0.0354	0.4643	0.4885	0.0046	0.0000	0.0000	0.0000	0.0000	0.0135	0.0270	0.3514	0.6081	0.0000	0.0000	0.0000	0.0000	
6	0.0023	0.0023	0.0234	0.2609	0.6659	0.0444	0.0008	0.0000	0.0000	0.0000	0.0000	0.0000	0.2879	0.6515	0.0606	0.0000	0.0000	0.0000	
7	0.0009	0.0019	0.0168	0.1551	0.7206	0.1037	0.0000	0.0009	0.0000	0.0000	0.0000	0.0130	0.1688	0.6234	0.1948	0.0000	0.0000	0.0000	
8	0.0000	0.0033	0.0098	0.0963	0.6740	0.2155	0.0011	0.0000	0.0000	0.0000	0.0000	0.0208	0.0625	0.5625	0.3542	0.0000	0.0000	0.0000	
9	0.0013	0.0000	0.0026	0.0480	0.5071	0.4332	0.0065	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4324	0.5676	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0033	0.0183	0.4027	0.5441	0.0316	0.0000	0.0000	0.0000	0.0000	0.0000	0.0417	0.2917	0.6667	0.0000	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0183	0.2633	0.6654	0.0530	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2778	0.6667	0.0556	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0080	0.2154	0.6809	0.0931	0.0027	0.0000	0.0000	0.0000	0.0000	0.0333	0.1667	0.6000	0.2000	0.0000	0.0000	
13	0.0000	0.0000	0.0024	0.0095	0.1718	0.6683	0.1456	0.0024	0.0000	0.0000	0.0000	0.0000	0.0526	0.2105	0.5789	0.1579	0.0000	0.0000	
14	0.0026	0.0000	0.0026	0.0128	0.0870	0.6547	0.2404	0.0000	0.0000	0.0000	0.0000	0.0000	0.0714	0.1071	0.4286	0.3929	0.0000	0.0000	
15	0.0024	0.0000	0.0024	0.0024	0.0558	0.6141	0.3204	0.0024	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6538	0.3077	0.0385	0.0000	
16	0.0000	0.0000	0.0000	0.0028	0.0386	0.4573	0.4766	0.0248	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.6522	0.3043	0.0000	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0337	0.4233	0.4816	0.0583	0.0031	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.6000	0.0000	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0145	0.3865	0.5217	0.0773	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4348	0.4348	0.1304	0.0000	
19	0.0000	0.0000	0.0056	0.0000	0.0112	0.3464	0.4413	0.1844	0.0112	0.0000	0.0000	0.0000	0.0000	0.0000	0.3846	0.3846	0.1538	0.0769	
20	0.0000	0.0000	0.0000	0.0048	0.0190	0.2571	0.4905	0.1952	0.0333	0.0000	0.0000	0.0000	0.0000	0.0714	0.2857	0.4286	0.1429	0.0714	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2474	0.5158	0.2053	0.0316	0.0000	0.0000	0.0000	0.0000	0.1111	0.1111	0.3333	0.4444	0.0000	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1634	0.4837	0.2876	0.0654	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.3333	0.3333	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0826	0.5596	0.2661	0.0917	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.1262	0.4369	0.2816	0.1553	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0588	0.4706	0.3824	0.0882	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.3333	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0625	0.3750	0.3438	0.2188	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0244	0.2439	0.3659	0.3659	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0244	0.2195	0.4146	0.3415	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0270	0.3784	0.1081	0.4865	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1818	0.1364	0.6818	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1333	0.3333	0.5333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	

TABLE B-6 (Continued)

3600 SN

2490 SH

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0410	0.1187	0.8403	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0157	0.0472	0.4587	0.4626	0.0157	0.0000	0.0000	0.0000	0.0000
4	0.0500	0.1183	0.8316	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0064	0.0339	0.2627	0.6144	0.0826	0.0000	0.0000	0.0000	0.0000
5	0.0500	0.0675	0.8426	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0103	0.0034	0.0897	0.7000	0.1931	0.0034	0.0000	0.0000	0.0000
6	0.1512	0.1221	0.7267	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0039	0.0000	0.0667	0.6471	0.2745	0.0039	0.0000	0.0000	0.0000
7	0.2130	0.0648	0.7222	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0090	0.0090	0.0511	0.5255	0.3964	0.0090	0.0000	0.0000	0.0000
8	0.1967	0.1639	0.6393	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0062	0.0062	0.0526	0.3344	0.5356	0.0062	0.0000	0.0000	0.0000
9	0.0857	0.2571	0.6571	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0043	0.0433	0.2771	0.6061	0.0093	0.0000	0.0000	0.0000
10	0.1765	0.1765	0.6471	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0217	0.1522	0.5942	0.2246	0.0072	0.0000	0.0000
11	0.3333	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0094	0.0986	0.3991	0.4742	0.0188	0.0000	0.0000
12	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0056	0.0667	0.4722	0.4167	0.0389	0.0000	0.0000
13	0.2000	0.0000	0.8000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0273	0.0492	0.3825	0.5082	0.0328	0.0000	0.0000
14	0.0000	0.3333	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0055	0.0497	0.4309	0.4309	0.0829	0.0000	0.0000
15	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0355	0.2485	0.5562	0.1479	0.0118	0.0000
16	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0067	0.0201	0.1745	0.6443	0.1409	0.0134	0.0000
17	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0183	0.0732	0.6402	0.2012	0.0610	0.0061
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0052	0.1458	0.6406	0.1667	0.0417	0.0000
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0213	0.0957	0.5851	0.2340	0.0638	0.0000
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0412	0.6289	0.1959	0.1134	0.0206
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4787	0.2766	0.1915	0.0213
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0319	0.3784	0.3243	0.2162	0.0811
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3462	0.2308	0.2308	0.1923
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2174	0.3043	0.2609	0.2174
25	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3158	0.3684	0.3158
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0909	0.4545	0.3636
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.3333	0.5833
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.6000
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0909	0.0909	0.1818	0.6364
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.8750

TABLE B-6 (Continued)

2000 SK

0250 SM

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0133	0.0301	0.2764	0.5899	0.0877	0.0027	0.0000	0.0000	0.0000	0.0157	0.0367	0.2572	0.5906	0.0997	0.0000	0.0000	0.0000	0.0000
4	0.0038	0.0132	0.1255	0.5636	0.2922	0.0038	0.0009	0.0000	0.0000	0.0133	0.0067	0.1400	0.4733	0.3667	0.0000	0.0000	0.0000	0.0000
5	0.0013	0.0013	0.0408	0.4711	0.4789	0.0039	0.0026	0.0000	0.0000	0.0102	0.0102	0.0646	0.2721	0.6361	0.0068	0.0000	0.0000	0.0000
6	0.0000	0.0040	0.0291	0.2761	0.6584	0.0211	0.0013	0.0000	0.0000	0.0051	0.0000	0.0205	0.1538	0.7744	0.0462	0.0000	0.0000	0.0000
7	0.0018	0.0018	0.0235	0.1859	0.6968	0.0804	0.0000	0.0018	0.0000	0.0000	0.0000	0.0161	0.1210	0.6210	0.2419	0.0000	0.0000	0.0000
8	0.0020	0.0041	0.0163	0.1321	0.5732	0.2724	0.0000	0.0000	0.0000	0.0055	0.0000	0.0110	0.0824	0.5055	0.3956	0.0000	0.0000	0.0000
9	0.0021	0.0041	0.0083	0.0766	0.4720	0.3488	0.0021	0.0000	0.0000	0.0000	0.0000	0.0189	0.0629	0.3585	0.5535	0.0063	0.0000	0.0000
10	0.0000	0.0000	0.0028	0.0028	0.0279	0.3370	0.6017	0.0306	0.0000	0.0000	0.0000	0.0265	0.0531	0.2301	0.6460	0.0442	0.0000	0.0000
11	0.0000	0.0000	0.0041	0.0186	0.1636	0.6625	0.1491	0.0021	0.0000	0.0000	0.0000	0.0109	0.0217	0.1196	0.7935	0.0543	0.0000	0.0000
12	0.0025	0.0000	0.0000	0.0000	0.0275	0.1650	0.6775	0.1275	0.0000	0.0000	0.0000	0.0127	0.0127	0.1392	0.7468	0.0886	0.0000	0.0000
13	0.0025	0.0000	0.0000	0.0000	0.0099	0.1464	0.6179	0.2233	0.0000	0.0000	0.0000	0.0000	0.0000	0.0805	0.8046	0.1149	0.0000	0.0000
14	0.0000	0.0000	0.0062	0.0093	0.1331	0.5604	0.2786	0.0124	0.0000	0.0000	0.0000	0.0000	0.0141	0.0563	0.6338	0.2958	0.0000	0.0000
15	0.0000	0.0000	0.0000	0.0047	0.1023	0.4930	0.3721	0.0279	0.0000	0.0000	0.0000	0.0000	0.0000	0.0714	0.5429	0.3857	0.0000	0.0000
16	0.0000	0.0000	0.0000	0.0000	0.0000	0.0686	0.4229	0.0857	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6129	0.3548	0.0323	0.0000
17	0.0000	0.0000	0.0000	0.0000	0.0000	0.0415	0.3983	0.3817	0.0041	0.0000	0.0000	0.0000	0.0217	0.0435	0.4130	0.4783	0.0435	0.0000
18	0.0000	0.0000	0.0000	0.0000	0.0226	0.2805	0.4344	0.2443	0.0181	0.0000	0.0000	0.0000	0.0000	0.0270	0.3243	0.4865	0.1351	0.0270
19	0.0000	0.0000	0.0000	0.0045	0.0227	0.3182	0.4273	0.1955	0.0318	0.0000	0.0000	0.0000	0.0000	0.0435	0.5652	0.2609	0.0870	0.0435
20	0.0000	0.0000	0.0000	0.0146	0.0293	0.2488	0.4244	0.2195	0.0634	0.0000	0.0000	0.0000	0.0000	0.0385	0.2308	0.5385	0.1923	0.0000
21	0.0000	0.0000	0.0000	0.0000	0.0138	0.2442	0.4101	0.2811	0.0507	0.0000	0.0000	0.0000	0.0000	0.0303	0.3030	0.4848	0.0909	0.0000
22	0.0000	0.0000	0.0000	0.0094	0.0000	0.1792	0.3868	0.3491	0.0755	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3077	0.5385	0.0769
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0822	0.4384	0.2603	0.2192	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5556	0.0000
24	0.0000	0.0000	0.0175	0.0000	0.0175	0.0702	0.3333	0.3158	0.2456	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.5833	0.2500	0.0833
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.1081	0.4054	0.2703	0.2162	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.1053	0.2632	0.1579	0.4737	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.7333	0.0667	0.2000
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.2857	0.5714	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3636	0.3636	0.1818
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0500	0.2750	0.2250	0.4500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.5833	0.0000	0.3333
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.3333	0.5556	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.3333	0.0000
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3000	0.3000	0.4000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3043	0.3478	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.5000	0.2500	0.1250
	0.0000	0.0000	0.0000	0.0000	0.0000	0.0526	0.0000	0.1579	0.7895	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000

TABLE B-6 (Continued)

0400 ST										7200 TD									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0053	0.0107	0.1132	0.5570	0.3137	0.0000	0.0000	0.0000	0.0000	0.0000	0.0192	0.6346	0.3462	0.0000	0.0000	0.0000	0.0000	0.0000	
4	0.0047	0.0071	0.0593	0.3108	0.6180	0.0000	0.0000	0.0000	0.0000	0.0000	0.0147	0.5074	0.4118	0.0662	0.0000	0.0000	0.0000	0.0000	
5	0.0026	0.0013	0.0290	0.1871	0.7325	0.0474	0.0000	0.0000	0.0000	0.0213	0.0213	0.2979	0.4468	0.2128	0.0000	0.0000	0.0000	0.0000	
6	0.0000	0.0013	0.0066	0.1273	0.6522	0.2126	0.0000	0.0000	0.0000	0.0000	0.0000	0.1364	0.3182	0.5455	0.0000	0.0000	0.0000	0.0000	
7	0.0000	0.0022	0.0129	0.0625	0.4569	0.4655	0.0000	0.0000	0.0000	0.0000	0.0000	0.0357	0.1071	0.8571	0.0000	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0072	0.0411	0.3333	0.6135	0.0048	0.0000	0.0000	0.0000	0.0000	0.0227	0.2273	0.7045	0.0455	0.0000	0.0000	0.0000	
9	0.0000	0.0054	0.0080	0.0295	0.1984	0.7131	0.0456	0.0000	0.0000	0.0000	0.0000	0.0417	0.1458	0.6667	0.1458	0.0000	0.0000	0.0000	
10	0.0032	0.0000	0.0000	0.0162	0.1071	0.6526	0.2208	0.0000	0.0000	0.0000	0.0313	0.0000	0.0313	0.9063	0.0313	0.0000	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0088	0.0088	0.0965	0.3728	0.0000	0.0000	0.0000	0.0000	0.0000	0.0370	0.4074	0.5556	0.0000	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0000	0.0000	0.0839	0.4965	0.4196	0.0000	0.0000	0.0000	0.0000	0.0000	0.2632	0.7368	0.0000	0.0000	0.0000	
13	0.0000	0.0000	0.0000	0.0000	0.0000	0.0423	0.3380	0.5704	0.0000	0.0000	0.0000	0.0000	0.0370	0.2593	0.6667	0.0370	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0088	0.0263	0.3158	0.5263	0.1228	0.0000	0.0000	0.0000	0.0000	0.0000	0.0833	0.8333	0.0833	0.0000	0.0000	
15	0.0000	0.0000	0.0000	0.0095	0.0286	0.1905	0.7143	0.0571	0.0000	0.0000	0.0000	0.0000	0.0000	0.0400	0.8400	0.1200	0.0000	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0217	0.1630	0.6413	0.1522	0.0217	0.0000	0.0000	0.0000	0.0000	0.0313	0.6563	0.2813	0.0313	0.0000	
17	0.0000	0.0000	0.0000	0.0119	0.0238	0.1786	0.4048	0.3571	0.0238	0.0000	0.0000	0.0000	0.0000	0.0250	0.6500	0.3000	0.0250	0.0000	
18	0.0000	0.0000	0.0000	0.0000	0.0000	0.0215	0.0753	0.3656	0.0430	0.0000	0.0000	0.0000	0.0000	0.0455	0.6364	0.1818	0.1364	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0897	0.3205	0.0769	0.0000	0.0000	0.0000	0.0000	0.0800	0.5600	0.2800	0.0800	0.0000	
20	0.0000	0.0000	0.0000	0.0000	0.0000	0.0938	0.3281	0.4688	0.1094	0.0000	0.0000	0.0000	0.0000	0.0000	0.5385	0.3077	0.1538	0.0000	
21	0.0000	0.0000	0.0000	0.0000	0.0189	0.1132	0.2642	0.4717	0.1321	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.4167	0.3333	0.0833	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.2200	0.5200	0.1600	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.6250	0.1250	0.0000	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0667	0.3333	0.3333	0.2667	0.0000	0.0000	0.0000	0.0000	0.0000	0.4000	0.4000	0.4000	0.2000	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1739	0.3478	0.4783	0.0000	0.0000	0.0000	0.0000	0.0000	0.2000	0.4000	0.4000	0.0000	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1333	0.4667	0.4000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.5000	0.0000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.2500	0.6250	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.0000	0.6667	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.1429	0.0000	0.4286	0.4286	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2857	0.0000	0.7143	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.6667	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	



TABLE B-6 (Continued)

0500 TM										5800 UT									
LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9	
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000	
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000	
3	0.0050	0.0270	0.2995	0.6361	0.0322	0.0000	0.0000	0.0000	0.0000	0.0110	0.0386	0.4545	0.4711	0.0248	0.0000	0.0000	0.0000	0.0000	
4	0.0083	0.0207	0.1307	0.6245	0.2137	0.0021	0.0000	0.0000	0.0000	0.0059	0.0177	0.2212	0.5369	0.2183	0.0000	0.0000	0.0000	0.0000	
5	0.0000	0.0058	0.0609	0.4493	0.4812	0.0029	0.0000	0.0000	0.0000	0.0000	0.0104	0.0992	0.3812	0.5065	0.0026	0.0000	0.0000	0.0000	
6	0.0000	0.0000	0.0082	0.2131	0.7664	0.0123	0.0000	0.0000	0.0000	0.0000	0.0049	0.0534	0.1893	0.7379	0.0097	0.0049	0.0000	0.0000	
7	0.0000	0.0000	0.0268	0.2146	0.6054	0.1533	0.0000	0.0000	0.0000	0.0000	0.0000	0.0128	0.1154	0.7628	0.1090	0.0000	0.0000	0.0000	
8	0.0000	0.0000	0.0075	0.1391	0.4850	0.3684	0.0000	0.0000	0.0000	0.0000	0.0000	0.0119	0.0714	0.4524	0.4643	0.0000	0.0000	0.0000	
9	0.0042	0.0084	0.0000	0.0591	0.3671	0.5485	0.0127	0.0000	0.0000	0.0000	0.0000	0.0115	0.0575	0.4368	0.4943	0.0000	0.0000	0.0000	
10	0.0000	0.0000	0.0000	0.0201	0.2412	0.7035	0.0352	0.0000	0.0000	0.0000	0.0000	0.0175	0.0175	0.3684	0.5614	0.0351	0.0000	0.0000	
11	0.0000	0.0000	0.0000	0.0361	0.1024	0.7169	0.1446	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2319	0.6957	0.0725	0.0000	0.0000	
12	0.0000	0.0000	0.0000	0.0174	0.0783	0.6522	0.2522	0.0000	0.0000	0.0000	0.0000	0.0000	0.0154	0.1692	0.6615	0.1385	0.0154	0.0000	
13	0.0000	0.0000	0.0000	0.0256	0.0256	0.6538	0.2949	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1695	0.7119	0.1186	0.0000	0.0000	
14	0.0000	0.0000	0.0000	0.0078	0.0391	0.5625	0.3750	0.0156	0.0000	0.0000	0.0000	0.0000	0.0000	0.1000	0.6714	0.2143	0.0143	0.0000	
15	0.0000	0.0000	0.0000	0.0000	0.0098	0.6176	0.3725	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0741	0.6296	0.2840	0.0123	0.0000	
16	0.0000	0.0000	0.0000	0.0000	0.0152	0.6212	0.3636	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1154	0.5769	0.2500	0.0577	0.0000	
17	0.0000	0.0000	0.0000	0.0000	0.0405	0.4054	0.5405	0.0135	0.0000	0.0000	0.0000	0.0000	0.0000	0.0702	0.4737	0.3509	0.0702	0.0175	
18	0.0000	0.0000	0.0000	0.0000	0.0244	0.3415	0.4878	0.1463	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5526	0.2895	0.1579	0.0000	
19	0.0000	0.0000	0.0000	0.0000	0.0192	0.2115	0.5769	0.1731	0.0192	0.0000	0.0000	0.0000	0.0000	0.0455	0.5000	0.2424	0.1515	0.0606	
20	0.0000	0.0000	0.0000	0.0000	0.0161	0.3387	0.4355	0.1935	0.0161	0.0000	0.0000	0.0000	0.0000	0.0000	0.4314	0.3529	0.1569	0.0588	
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.4091	0.2727	0.0455	0.0000	0.0000	0.0000	0.0000	0.0000	0.3030	0.3333	0.3030	0.0606	
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.0857	0.4571	0.3143	0.1429	0.0000	0.0000	0.0000	0.0000	0.0667	0.2000	0.4000	0.1333	0.2000	
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0645	0.3226	0.3548	0.2581	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.3333	0.1667	
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0417	0.4583	0.2917	0.2083	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.0833	0.3333	0.4167	
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.2500	0.4167	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3333	0.2000	0.4667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1250	0.1250	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2500	0.0000	0.7500	
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2727	0.6364	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.5000	
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.3333	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.5000	0.0000	0.5000	
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1667	0.5000	0.3333	
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.1111	0.0000	0.8889	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	

TABLE B-6 (Continued)

1700 YN

LOS	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
1	0.4396	0.3532	0.1624	0.0432	0.0010	0.0004	0.0001	0.0000	0.0000
2	0.0293	0.1634	0.5810	0.2187	0.0069	0.0006	0.0001	0.0000	0.0000
3	0.0040	0.0190	0.3096	0.5740	0.0863	0.0040	0.0016	0.0000	0.0000
4	0.0045	0.0090	0.1228	0.5978	0.2614	0.0037	0.0007	0.0000	0.0000
5	0.0026	0.0026	0.0460	0.4248	0.5169	0.0070	0.0000	0.0000	0.0000
6	0.0021	0.0031	0.0374	0.2555	0.6615	0.0395	0.0010	0.0000	0.0000
7	0.0000	0.0024	0.0171	0.1812	0.6426	0.1530	0.0037	0.0000	0.0000
8	0.0027	0.0041	0.0068	0.1096	0.5781	0.2945	0.0041	0.0000	0.0000
9	0.0016	0.0016	0.0082	0.0474	0.4330	0.4935	0.0131	0.0016	0.0000
10	0.0000	0.0000	0.0046	0.0410	0.2711	0.6469	0.0342	0.0023	0.0000
11	0.0000	0.0000	0.0000	0.0415	0.2445	0.6419	0.0699	0.0000	0.0022
12	0.0000	0.0024	0.0000	0.0190	0.1706	0.6493	0.1588	0.0000	0.0000
13	0.0021	0.0021	0.0000	0.0185	0.1235	0.6872	0.1667	0.0000	0.0000
14	0.0000	0.0000	0.0000	0.0045	0.0750	0.6659	0.2432	0.0114	0.0000
15	0.0000	0.0000	0.0026	0.0026	0.0599	0.5521	0.3594	0.0234	0.0000
16	0.0030	0.0000	0.0000	0.0030	0.0544	0.4653	0.4562	0.0181	0.0000
17	0.0000	0.0000	0.0035	0.0071	0.0355	0.4220	0.4113	0.1170	0.0035
18	0.0000	0.0000	0.0000	0.0000	0.0130	0.3478	0.5087	0.1217	0.0087
19	0.0000	0.0000	0.0000	0.0000	0.0106	0.3138	0.5319	0.1277	0.0160
20	0.0000	0.0000	0.0000	0.0000	0.0169	0.2978	0.4719	0.1798	0.0337
21	0.0000	0.0000	0.0000	0.0000	0.0000	0.2320	0.4640	0.2560	0.0480
22	0.0000	0.0000	0.0000	0.0000	0.0000	0.1739	0.4239	0.3261	0.0761
23	0.0000	0.0000	0.0000	0.0000	0.0000	0.0959	0.4658	0.2329	0.2055
24	0.0000	0.0000	0.0000	0.0000	0.0000	0.0746	0.4627	0.2836	0.1791
25	0.0000	0.0000	0.0000	0.0000	0.0000	0.0769	0.2923	0.3692	0.2615
26	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.4694	0.3469	0.1837
27	0.0000	0.0000	0.0000	0.0000	0.0000	0.0690	0.3793	0.2414	0.3103
28	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3043	0.4348	0.2609
29	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3667	0.1667	0.4667
30	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.2667	0.3333	0.4000
31	0.0000	0.0000	0.0000	0.0000	0.0000	0.0435	0.0870	0.1304	0.7391



TABLE B-7

DISTRIBUTION OF PRIOR SERVICE  
ACCESSIONS BY LOS <= 9 (FY 1986)

LOS	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.1973	0.0645	0.0833	0.0123	0.0571	0.0390	0.0078	0.0000	0.0181	0.0462	0.0417	0.0121	0.0000	0.2500
2	0.0807	0.0258	0.1250	0.0270	0.0286	0.0260	0.0234	0.0000	0.0361	0.0308	0.0208	0.0121	0.0044	0.0000
3	0.0897	0.0065	0.0000	0.0123	0.0000	0.0260	0.0000	0.0000	0.0144	0.0000	0.0208	0.0040	0.0044	0.0000
4	0.2601	0.1290	0.0833	0.1029	0.2000	0.1558	0.2188	0.1250	0.1047	0.1538	0.1250	0.2024	0.1491	0.2500
5	0.2646	0.4581	0.1250	0.5368	0.3714	0.3896	0.4531	0.5750	0.5090	0.5231	0.3125	0.5304	0.4693	0.2500
6	0.0314	0.0645	0.2917	0.0907	0.1143	0.1425	0.0703	0.1500	0.0650	0.1538	0.1042	0.0891	0.1404	0.0000
7	0.0359	0.1226	0.1250	0.0515	0.0286	0.0909	0.0938	0.0750	0.1083	0.0154	0.2083	0.0729	0.0921	0.1250
8	0.0224	0.0839	0.0000	0.0882	0.1143	0.0649	0.0703	0.0250	0.0614	0.0000	0.1042	0.0324	0.0482	0.1250
9	0.0179	0.0452	0.1667	0.0784	0.0857	0.0649	0.0625	0.0500	0.0830	0.0769	0.0625	0.0445	0.0921	0.0000

LOS	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.1429	0.0000	0.0000	0.1818	0.1034	0.0000	0.1111	0.2857	0.0385	0.0000	0.0250	0.0000	0.0000	0.0118
2	0.1429	0.0000	0.0000	0.1818	0.0690	0.0417	0.1556	0.1429	0.0385	0.0000	0.0000	0.4000	0.0100	0.0237
3	0.0714	0.0000	0.0000	0.0000	0.0690	0.0000	0.0222	0.0000	0.0000	0.0000	0.0250	0.0000	0.0100	0.0118
4	0.1429	0.0000	0.2500	0.0000	0.0000	0.0000	0.1000	0.2143	0.1154	0.1429	0.0500	0.0000	0.0657	0.1657
5	0.2143	0.5714	0.1250	0.5455	0.2414	0.7083	0.0667	0.1429	0.5385	0.2857	0.0500	0.4000	0.5622	0.6568
6	0.0714	0.1429	0.0000	0.0000	0.2414	0.0417	0.3444	0.0714	0.1538	0.2857	0.4500	0.0000	0.1194	0.0473
7	0.0714	0.1429	0.5000	0.0909	0.1379	0.0833	0.0667	0.0714	0.1154	0.0000	0.2500	0.2000	0.1443	0.0237
8	0.0000	0.0000	0.1250	0.0000	0.1034	0.0417	0.0444	0.0000	0.0000	0.1429	0.1000	0.0000	0.0348	0.0296
9	0.1429	0.1429	0.0000	0.0000	0.0345	0.0833	0.0889	0.0714	0.0000	0.1429	0.0500	0.0000	0.0498	0.0296

TABLE B-7 (Continued)

LOS	5350 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.1667	0.0128	0.0526	0.3292	0.0137	0.0176	0.0000	0.1208	0.0245	0.0769	0.0000	0.1250	0.0000	0.0000
2	0.1923	0.0128	0.1579	0.1523	0.0137	0.0235	0.0000	0.1946	0.0070	0.0000	0.0000	0.0000	0.0000	0.0000
3	0.0128	0.0128	0.0000	0.0576	0.0000	0.0000	0.0000	0.0470	0.0070	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.1410	0.0641	0.0526	0.2016	0.1233	0.1882	0.0000	0.1342	0.1923	0.0769	0.0833	0.2500	0.0000	0.0909
5	0.0769	0.1154	0.4211	0.1523	0.3288	0.5529	0.4444	0.2215	0.5594	0.4615	0.4167	0.3750	0.0000	0.0909
6	0.2051	0.0128	0.0000	0.0412	0.0411	0.0824	0.1667	0.0738	0.0839	0.0769	0.1667	0.0000	0.0000	0.1818
7	0.1282	0.4615	0.2105	0.0412	0.3699	0.0294	0.3333	0.0872	0.0559	0.2308	0.1667	0.1250	1.0000	0.1818
8	0.0256	0.1410	0.1053	0.0123	0.0685	0.0353	0.0556	0.0671	0.0350	0.0000	0.1667	0.0000	0.0000	0.3636
9	0.0513	0.1667	0.0000	0.0123	0.0411	0.0706	0.0000	0.0537	0.0350	0.0769	0.0000	0.1250	0.0000	0.0909

LOS	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.0000	0.0093	0.1667	0.0286	0.1048	0.0000	1.0000	0.0538	0.0000	0.0000	0.0000	0.0000	0.0000	0.0448
2	0.0000	0.0031	0.2222	0.0286	0.0286	0.0000	0.0000	0.0269	0.0000	0.0000	0.0417	0.0667	0.0000	0.0000
3	0.0000	0.0155	0.0556	0.0000	0.0286	0.0000	0.0000	0.0161	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4	0.2857	0.0836	0.1111	0.2571	0.1524	0.1000	0.0000	0.0914	0.0625	0.1818	0.0417	0.0000	0.0000	0.0448
5	0.5714	0.4582	0.2222	0.4571	0.2857	0.3000	0.0000	0.5054	0.4375	0.7273	0.0833	0.6667	1.0000	0.4925
6	0.0000	0.1176	0.1111	0.0857	0.1429	0.3000	0.0000	0.1505	0.0625	0.0909	0.4583	0.6667	0.0000	0.0746
7	0.0000	0.1331	0.0000	0.0286	0.0095	0.1000	0.0000	0.0161	0.0625	0.0000	0.2083	0.1333	0.0000	0.0896
8	0.0000	0.0774	0.0556	0.0000	0.0667	0.1000	0.0000	0.0484	0.2500	0.0000	0.1667	0.0667	0.0000	0.1343
9	0.1429	0.1022	0.0556	0.1143	0.1810	0.1000	0.0000	0.0914	0.1250	0.0000	0.0000	0.0000	0.0000	0.1194

LOS	7000 PR	0200 CM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN
1	0.0196	0.0800	0.0652	0.0435	0.4521	0.0000	0.0860	0.0508	0.0267	0.0000	0.0000	0.2000	0.0336
2	0.0000	0.0133	0.0652	0.0000	0.1242	0.0000	0.0323	0.0000	0.0133	0.0000	0.0000	0.2000	0.0067
3	0.0000	0.0267	0.0217	0.0435	0.0466	0.0476	0.0000	0.0339	0.0667	0.0000	0.0167	0.0000	0.0134
4	0.1961	0.1867	0.1304	0.0870	0.1078	0.1429	0.1398	0.1525	0.1333	0.0000	0.1167	0.1778	0.1208
5	0.4118	0.5067	0.4710	0.6087	0.1808	0.4762	0.3656	0.4746	0.3733	0.5000	0.7167	0.1333	0.5436
6	0.1569	0.0667	0.0652	0.0870	0.0320	0.0552	0.1075	0.1356	0.0800	0.2500	0.0167	0.2000	0.0738
7	0.0980	0.0267	0.0507	0.0435	0.0256	0.0476	0.0968	0.0169	0.2533	0.0000	0.0000	0.0444	0.0738
8	0.0588	0.0400	0.0870	0.0870	0.0128	0.1190	0.1075	0.0508	0.0267	0.2500	0.0167	0.0000	0.0403
9	0.0588	0.0533	0.0435	0.0000	0.0183	0.0714	0.0645	0.0847	0.0267	0.0000	0.1167	0.0444	0.0940

TABLE B-8

ACTIVE DUTY REQUIREMENTS BY  
FISCAL YEAR, RATING AND PAYGRADE  
(INCLUDES TAR PROGRAM)

Requirements for 1986

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3013.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	20621.	1171.	452.	5869.	117.	962.	1691.	366.	4026.	786.	620.	1076.	1561.	140.
4	0.	2499.	707.	6669.	462.	1295.	1921.	827.	5688.	691.	1129.	3375.	3519.	0.
5	0.	1682.	857.	7711.	411.	1834.	1747.	959.	8251.	1116.	1251.	2934.	2461.	0.
6	0.	1168.	666.	6495.	395.	913.	1544.	558.	5251.	865.	855.	2139.	1984.	0.
7	0.	571.	230.	2343.	159.	376.	475.	169.	1955.	349.	356.	1284.	828.	0.
8	0.	125.	61.	967.	47.	80.	112.	76.	927.	102.	68.	385.	292.	0.
9	0.	73.	26.	399.	19.	49.	66.	21.	396.	29.	43.	167.	197.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	142.	216.	2.	318.	351.	263.	984.	1286.	428.	13.	745.	8.	1692.	791.
4	269.	226.	655.	543.	506.	476.	978.	964.	713.	98.	1100.	1049.	7193.	2738.
5	353.	326.	897.	546.	519.	623.	1047.	660.	815.	144.	1085.	871.	5389.	2226.
6	230.	323.	500.	418.	439.	565.	829.	408.	746.	104.	792.	663.	4252.	2065.
7	140.	130.	188.	164.	225.	268.	362.	187.	165.	16.	391.	216.	1795.	823.
8	37.	34.	64.	48.	64.	62.	116.	56.	88.	8.	82.	37.	664.	170.
9	18.	17.	27.	29.	29.	34.	45.	21.	23.	2.	29.	21.	200.	84.

TABLE B-8 (Continued)

Requirements for 1986

IPAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	3325.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	778.	340.	340.	18205.	526.	873.	129.	7987.	1060.	203.	104.	84.	0.	0.
4	898.	6610.	695.	0.	4240.	2144.	1000.	6737.	3756.	322.	288.	127.	0.	0.
5	902.	5608.	563.	0.	3338.	2009.	820.	5316.	3377.	531.	252.	152.	221.	13.
6	715.	4400.	666.	0.	3210.	2043.	649.	3573.	2749.	423.	188.	98.	189.	1182.
7	234.	1811.	406.	0.	1395.	1065.	401.	2211.	1254.	125.	108.	47.	125.	518.
8	61.	496.	67.	0.	465.	172.	97.	471.	273.	34.	41.	1.	25.	137.
9	32.	206.	25.	0.	201.	74.	20.	209.	133.	15.	9.	2.	12.	77.

IPAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	35.	2189.	53.	308.	3285.	46.	0.	2872.	260.	172.	222.	85.	15.	950.
4	70.	9549.	212.	790.	4711.	245.	0.	3519.	340.	565.	500.	291.	40.	1595.
5	67.	6435.	141.	1028.	4274.	197.	0.	2523.	474.	184.	519.	332.	49.	2064.
6	54.	6148.	116.	643.	3314.	198.	909.	2029.	438.	192.	359.	300.	55.	2067.
7	40.	2168.	55.	227.	1311.	94.	599.	614.	174.	65.	129.	108.	21.	870.
8	7.	893.	27.	45.	360.	37.	224.	324.	56.	17.	29.	25.	0.	237.
9	2.	444.	14.	25.	174.	18.	124.	73.	19.	20.	15.	6.	0.	172.

IPAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	15800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	6090.	0.	0.	0.	0.	0.	0.	0.	0.	12778.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	482.	512.	3246.	319.	44132.	983.	1114.	644.	1403.	6.	468.	502.	2742.	144371.
4	587.	1174.	4961.	283.	0.	1455.	2361.	1028.	2808.	46.	1180.	609.	3209.	115235.
5	764.	967.	4634.	268.	0.	1238.	2680.	833.	1996.	61.	960.	823.	3621.	106919.
6	518.	955.	3082.	210.	0.	1024.	2131.	649.	2027.	34.	892.	575.	3118.	87289.
7	108.	731.	1385.	85.	0.	262.	1254.	236.	764.	14.	450.	184.	1433.	37246.
8	40.	133.	546.	22.	0.	107.	414.	50.	404.	3.	152.	75.	367.	11706.
9	21.	56.	153.	10.	0.	70.	166.	35.	94.	2.	61.	1.	137.	4991.

TABLE B-8 (Continued)

Requirements for 1987

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3193.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	21991.	1221.	479.	6100.	122.	984.	1749.	400.	4181.	811.	608.	1098.	1571.	142.
4	0.	2604.	713.	6831.	474.	1310.	1974.	864.	5870.	703.	1151.	3484.	3570.	0.
5	0.	1737.	875.	7948.	431.	1833.	1786.	979.	8557.	1179.	1274.	3034.	2534.	0.
6	0.	1193.	670.	6645.	398.	908.	1593.	578.	5428.	879.	862.	2189.	2031.	0.
7	0.	586.	232.	2426.	159.	380.	493.	193.	2088.	372.	371.	1336.	851.	0.
8	0.	*28.	65.	970.	49.	78.	123.	80.	934.	103.	70.	395.	294.	0.
9	0.	73.	26.	408.	19.	49.	64.	19.	409.	29.	43.	170.	203.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	142.	216.	2.	327.	341.	263.	1010.	1321.	449.	13.	751.	10.	1721.	823.
4	258.	222.	655.	557.	513.	485.	995.	1033.	704.	103.	1091.	1052.	7382.	2882.
5	346.	338.	914.	547.	549.	627.	1072.	675.	813.	144.	1091.	903.	5586.	2372.
6	229.	331.	504.	424.	460.	571.	868.	418.	750.	108.	854.	709.	4437.	2181.
7	140.	128.	188.	164.	229.	273.	388.	192.	155.	16.	392.	213.	1876.	862.
8	37.	34.	64.	49.	65.	65.	120.	60.	88.	8.	75.	36.	679.	182.
9	18.	17.	27.	29.	30.	34.	49.	22.	20.	2.	29.	21.	206.	93.

TABLE B-8 (Continued)

Requirements for 1987

PAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	3607.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	784.	340.	350.	17870.	533.	885.	145.	8237.	1106.	208.	104.	89.	0.	0.
4	924.	6768.	735.	0.	4364.	2208.	1062.	6906.	3863.	352.	289.	127.	0.	0.
5	933.	5811.	595.	0.	3493.	2085.	854.	5416.	3457.	543.	265.	152.	236.	45.
6	742.	4535.	694.	0.	3418.	2163.	707.	3733.	2852.	463.	190.	104.	195.	1317.
7	240.	1880.	417.	0.	1443.	1124.	419.	2251.	1315.	134.	103.	47.	128.	584.
8	66.	506.	71.	0.	479.	181.	100.	479.	278.	35.	39.	1.	25.	148.
9	33.	213.	25.	0.	210.	77.	25.	215.	136.	15.	8.	2.	12.	84.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	34.	2264.	54.	315.	3624.	46.	0.	3069.	255.	176.	222.	91.	17.	905.
4	71.	9642.	214.	825.	4875.	245.	0.	3638.	354.	577.	505.	315.	40.	1554.
5	69.	6649.	158.	1047.	4495.	197.	0.	2620.	470.	190.	514.	366.	50.	2013.
6	51.	6280.	120.	675.	3462.	198.	983.	2135.	454.	201.	364.	327.	56.	2086.
7	40.	2215.	58.	233.	1334.	94.	624.	628.	173.	66.	137.	113.	21.	899.
8	5.	919.	28.	47.	377.	37.	226.	342.	56.	17.	32.	27.	0.	241.
9	2.	446.	15.	23.	178.	18.	127.	75.	19.	19.	17.	4.	0.	168.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	5628.	0.	0.	0.	0.	0.	0.	0.	0.	12778.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	491.	531.	3303.	319.	43983.	1043.	1134.	671.	1431.	1.	494.	515.	2749.	147234.
4	597.	1218.	5080.	285.	0.	1530.	2437.	1050.	2972.	4.	1211.	650.	3190.	118157.
5	770.	990.	4743.	281.	0.	1260.	2759.	876.	2115.	8.	973.	837.	3688.	110169.
6	514.	985.	3156.	211.	0.	1007.	2196.	673.	2166.	5.	929.	595.	3198.	90358.
7	117.	737.	1408.	84.	0.	255.	1280.	237.	799.	2.	483.	199.	1453.	38477.
8	36.	138.	555.	24.	0.	112.	426.	50.	414.	0.	169.	75.	378.	11990.
9	21.	56.	155.	10.	0.	71.	168.	36.	98.	0.	62.	1.	125.	5078.

TABLE B-8 (Continued)

Requirements for 1988

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3140.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	22367.	1236.	488.	6071.	122.	995.	1756.	388.	4194.	819.	601.	1104.	1588.	142.
4	0.	2611.	735.	6807.	479.	1318.	1979.	858.	5875.	711.	1153.	3547.	3669.	0.
5	0.	1746.	898.	7941.	433.	1834.	1795.	976.	8574.	1189.	1262.	3092.	2600.	0.
6	0.	1206.	697.	6645.	410.	908.	1594.	575.	5448.	887.	864.	2248.	2063.	0.
7	0.	585.	237.	2424.	160.	380.	494.	193.	2097.	374.	371.	1380.	877.	0.
8	0.	133.	66.	972.	50.	78.	124.	80.	949.	104.	70.	401.	296.	0.
9	0.	73.	26.	409.	19.	50.	65.	19.	415.	29.	42.	173.	207.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	142.	216.	2.	331.	345.	263.	1008.	1309.	453.	13.	746.	10.	1740.	856.
4	262.	219.	692.	557.	521.	494.	1012.	1042.	700.	104.	1096.	1062.	7237.	2985.
5	346.	336.	914.	549.	561.	627.	1079.	690.	787.	142.	1095.	908.	5742.	2472.
6	234.	331.	517.	433.	468.	584.	881.	423.	754.	109.	855.	719.	4514.	2236.
7	142.	129.	188.	167.	237.	276.	388.	197.	158.	16.	396.	217.	1908.	893.
8	37.	35.	65.	49.	66.	65.	119.	61.	88.	8.	75.	36.	690.	183.
9	18.	17.	27.	30.	30.	34.	49.	22.	20.	2.	29.	21.	211.	100.

TABLE B-8 (Continued)

Requirements for 1988

PAY	15380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	1440 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	9.	0.	3675.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	795.	346.	354.	18378.	543.	890.	147.	8343.	1140.	209.	105.	89.	0.	0.
4	930.	6846.	781.	0.	4484.	2265.	1173.	7019.	3998.	358.	290.	128.	0.	0.
5	943.	5934.	618.	0.	3630.	2122.	891.	5502.	3534.	553.	270.	152.	239.	75.
6	748.	4560.	719.	0.	3476.	2203.	693.	3780.	2925.	480.	192.	106.	204.	1335.
7	242.	1896.	417.	0.	1457.	1130.	421.	2294.	1324.	144.	103.	47.	128.	599.
8	65.	509.	74.	0.	489.	182.	102.	484.	286.	35.	39.	1.	25.	153.
9	33.	218.	25.	0.	214.	80.	27.	217.	140.	15.	9.	2.	12.	85.

PAY	14700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	11400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	33.	2292.	54.	328.	3697.	46.	0.	3089.	255.	176.	222.	91.	16.	914.
4	68.	9775.	212.	833.	4923.	245.	0.	3754.	375.	587.	509.	315.	39.	1549.
5	67.	6610.	158.	1082.	4532.	198.	0.	2702.	497.	192.	517.	366.	50.	2007.
6	54.	6411.	120.	692.	3545.	198.	998.	2203.	490.	203.	372.	327.	58.	2056.
7	40.	2243.	58.	232.	1352.	94.	626.	663.	178.	68.	135.	113.	21.	906.
8	5.	937.	28.	47.	386.	37.	225.	348.	60.	17.	33.	28.	0.	242.
9	2.	452.	15.	25.	181.	18.	127.	81.	22.	18.	20.	4.	0.	170.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	10500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	5578.	0.	0.	0.	0.	0.	0.	0.	0.	12743.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	479.	545.	3332.	342.	45482.	1051.	1165.	683.	1426.	0.	497.	514.	2797.	150170.
4	588.	1247.	5160.	304.	0.	1542.	2482.	1074.	3139.	0.	1228.	664.	3216.	119825.
5	772.	1037.	4833.	283.	0.	1279.	2790.	906.	2203.	0.	975.	854.	3706.	111667.
6	515.	1007.	3188.	211.	0.	1005.	2246.	690.	2230.	0.	934.	602.	3220.	91599.
7	117.	758.	1426.	86.	0.	256.	1285.	246.	815.	0.	493.	197.	1479.	38973.
8	36.	138.	555.	25.	0.	114.	437.	50.	427.	0.	173.	75.	381.	12148.
9	21.	57.	155.	10.	0.	71.	171.	36.	99.	0.	63.	1.	122.	5155.



TABLE B-8 (Continued)

Requirements for 1989

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3166.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	22319.	1232.	494.	6043.	132.	1002.	1758.	388.	4210.	824.	607.	1132.	1591.	142.
4	0.	2702.	747.	6826.	500.	1336.	2008.	858.	5893.	718.	1159.	3654.	3659.	0.
5	0.	1781.	918.	7951.	449.	1862.	1816.	985.	8625.	1197.	1280.	3175.	2628.	0.
6	0.	1223.	701.	6676.	423.	917.	1617.	587.	5473.	891.	865.	2281.	2070.	0.
7	0.	592.	241.	2434.	165.	386.	501.	196.	2129.	377.	374.	1399.	886.	0.
8	0.	137.	66.	981.	52.	78.	126.	81.	965.	104.	72.	407.	300.	0.
9	0.	73.	27.	410.	20.	51.	66.	19.	417.	29.	42.	174.	209.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CIR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	143.	216.	2.	341.	350.	264.	1007.	1405.	470.	13.	761.	10.	1779.	926.
4	262.	219.	694.	557.	536.	508.	1052.	1084.	693.	104.	1100.	1084.	7633.	3087.
5	348.	336.	920.	551.	574.	632.	1115.	726.	791.	142.	1099.	926.	5888.	2548.
6	238.	331.	519.	443.	476.	589.	924.	445.	757.	112.	859.	737.	4572.	2263.
7	143.	130.	191.	169.	246.	278.	395.	207.	157.	16.	397.	230.	1943.	921.
8	37.	34.	65.	48.	67.	64.	122.	63.	89.	8.	75.	37.	702.	185.
9	18.	17.	27.	30.	30.	34.	49.	22.	20.	2.	29.	21.	218.	103.

TABLE B-8 (Continued)

Requirements for 1989

PAY	15380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	3709.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	792.	346.	367.	18530.	564.	917.	149.	9273.	1149.	211.	105.	90.	0.	0.
4	958.	6920.	828.	0.	4705.	2350.	1246.	7793.	4074.	373.	295.	131.	0.	0.
5	969.	6015.	634.	0.	3789.	2190.	929.	6094.	3608.	570.	274.	158.	259.	75.
6	769.	4639.	724.	0.	3594.	2245.	704.	4122.	2977.	500.	192.	108.	217.	1335.
7	244.	1927.	421.	0.	1489.	1154.	439.	2503.	1353.	147.	103.	48.	129.	599.
8	67.	517.	76.	0.	506.	183.	104.	535.	288.	36.	40.	1.	25.	155.
9	35.	229.	25.	0.	229.	81.	31.	236.	142.	15.	9.	2.	12.	87.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	14600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	34.	2293.	54.	329.	3816.	46.	0.	3177.	281.	181.	229.	90.	16.	930.
4	68.	9273.	214.	928.	5049.	245.	0.	3912.	380.	599.	514.	314.	39.	1587.
5	67.	6836.	158.	1108.	4557.	198.	0.	2844.	531.	195.	524.	366.	47.	2050.
6	57.	6488.	125.	709.	3611.	198.	1015.	2289.	499.	206.	384.	332.	59.	2077.
7	40.	2268.	60.	234.	1352.	94.	627.	682.	177.	68.	144.	113.	21.	926.
8	5.	949.	30.	49.	386.	37.	225.	353.	60.	17.	33.	28.	0.	245.
9	2.	465.	15.	27.	181.	18.	129.	84.	22.	18.	26.	4.	0.	173.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	5852.	0.	0.	0.	0.	0.	0.	0.	0.	13077.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	479.	561.	3409.	342.	45706.	1096.	1185.	703.	1473.	0.	502.	514.	2844.	152344.
4	591.	1276.	5238.	299.	0.	1583.	2524.	1110.	3215.	0.	1235.	699.	3251.	122389.
5	776.	1086.	4855.	285.	0.	1287.	2829.	928.	2255.	0.	980.	878.	3748.	114215.
6	521.	1031.	3242.	213.	0.	1021.	2280.	706.	2279.	0.	947.	615.	3237.	93256.
7	121.	760.	1459.	88.	0.	248.	1286.	246.	833.	0.	498.	197.	1483.	39684.
8	36.	138.	557.	25.	0.	115.	440.	50.	440.	0.	174.	77.	387.	12354.
9	21.	57.	156.	10.	0.	71.	172.	38.	99.	0.	64.	1.	119.	5253.

TABLE B-8 (Continued)

Requirements for 1990

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3166.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	22346.	1263.	494.	6063.	132.	999.	1748.	388.	4224.	832.	607.	1158.	1557.	143.
4	0.	2694.	747.	6822.	500.	1346.	2008.	858.	5905.	727.	1160.	3674.	3675.	0.
5	0.	1778.	918.	7941.	450.	1846.	1810.	990.	8548.	1209.	1288.	3188.	2643.	0.
6	0.	1223.	701.	6669.	427.	913.	1619.	587.	5481.	897.	863.	2287.	2087.	0.
7	0.	597.	241.	2430.	170.	385.	501.	196.	2129.	380.	371.	1419.	890.	0.
8	0.	137.	66.	983.	52.	78.	126.	81.	966.	104.	72.	411.	300.	0.
9	0.	73.	27.	410.	20.	51.	66.	19.	417.	29.	42.	175.	209.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	143.	216.	2.	351.	352.	264.	1009.	1416.	480.	13.	768.	10.	1783.	961.
4	263.	219.	696.	569.	538.	508.	1053.	1091.	705.	104.	1107.	1084.	7682.	3136.
5	350.	336.	932.	566.	584.	633.	1115.	735.	800.	144.	1102.	949.	5935.	2581.
6	238.	331.	523.	457.	485.	593.	923.	451.	766.	112.	860.	739.	4598.	2285.
7	143.	130.	192.	171.	258.	278.	397.	207.	158.	16.	398.	231.	1946.	937.
8	37.	34.	65.	48.	67.	64.	122.	64.	90.	8.	76.	37.	706.	185.
9	18.	17.	27.	30.	30.	34.	49.	22.	20.	2.	29.	21.	220.	102.

TABLE B-8 (Continued)

Requirements for 1990

PAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	3709.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	794.	346.	370.	18646.	564.	926.	151.	9773.	1165.	211.	105.	92.	0.	0.
4	961.	6654.	847.	0.	4813.	2400.	1333.	8075.	4133.	381.	296.	133.	0.	0.
5	972.	6061.	640.	0.	3874.	2222.	969.	6321.	3682.	577.	278.	161.	260.	75.
6	769.	4692.	732.	0.	3669.	2272.	725.	4284.	3034.	502.	192.	109.	217.	1347.
7	247.	1946.	427.	0.	1516.	1162.	454.	2604.	1377.	149.	103.	49.	129.	607.
8	67.	522.	76.	0.	518.	184.	107.	557.	288.	36.	40.	1.	25.	155.
9	35.	224.	25.	0.	235.	81.	34.	244.	144.	15.	9.	2.	12.	88.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	11080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	33.	2298.	54.	336.	3828.	46.	0.	3185.	281.	181.	229.	92.	17.	930.
4	69.	9839.	214.	848.	5067.	245.	0.	3963.	390.	600.	516.	321.	39.	1586.
5	69.	6814.	158.	1132.	4601.	198.	0.	2913.	533.	195.	538.	369.	47.	2052.
6	58.	6486.	125.	727.	3635.	198.	1033.	2335.	509.	207.	387.	336.	59.	2084.
7	40.	2277.	60.	240.	1352.	94.	627.	693.	179.	68.	144.	114.	21.	927.
8	5.	951.	30.	49.	389.	37.	225.	356.	60.	17.	33.	28.	0.	245.
9	2.	465.	15.	28.	184.	18.	129.	87.	22.	18.	32.	4.	0.	172.

PAY	7600 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	5852.	0.	0.	0.	0.	0.	0.	0.	0.	13077.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	479.	564.	3435.	342.	45795.	1098.	1186.	708.	1488.	0.	503.	514.	2882.	153369.
4	593.	1276.	5304.	299.	0.	1593.	2528.	1123.	3287.	0.	1235.	701.	3272.	123805.
5	782.	1125.	5018.	290.	0.	1294.	2857.	931.	2290.	0.	980.	887.	3772.	115408.
6	519.	1037.	3266.	214.	0.	1013.	2293.	711.	2322.	0.	954.	610.	3270.	94047.
7	123.	761.	1480.	88.	0.	248.	2293.	246.	839.	0.	495.	201.	1490.	40039.
8	36.	142.	558.	25.	0.	115.	442.	50.	444.	0.	174.	78.	387.	12431.
9	21.	57.	158.	10.	0.	71.	172.	38.	99.	0.	64.	1.	119.	5293.

TABLE B-8 (Continued)

Requirements for 1991

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AC	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	3166.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	350.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	22671.	1268.	500.	6079.	133.	1001.	1738.	390.	4242.	837.	609.	1167.	1591.	143.
4	0.	2737.	753.	6861.	504.	1362.	2010.	864.	5931.	729.	1163.	3718.	3693.	0.
5	0.	1799.	928.	7976.	453.	1850.	1815.	1002.	8700.	1211.	1290.	3222.	2648.	0.
6	0.	1236.	706.	6677.	429.	916.	1621.	592.	5507.	898.	865.	2299.	2095.	0.
7	0.	601.	242.	2432.	170.	386.	501.	197.	2130.	381.	373.	1419.	894.	0.
8	0.	137.	66.	983.	52.	78.	126.	81.	965.	104.	72.	412.	303.	0.
9	0.	73.	27.	411.	20.	51.	66.	19.	417.	29.	42.	176.	209.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	18300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	143.	217.	2.	363.	352.	266.	1011.	1419.	491.	14.	771.	16.	1805.	980.
4	263.	219.	704.	573.	541.	510.	1071.	1097.	705.	106.	1108.	1095.	7771.	3174.
5	352.	336.	963.	579.	589.	634.	1131.	740.	802.	145.	1102.	964.	5991.	2601.
6	238.	332.	529.	465.	487.	594.	935.	454.	777.	112.	862.	748.	4625.	2289.
7	143.	130.	195.	177.	260.	279.	405.	209.	160.	16.	399.	232.	1955.	942.
8	37.	34.	65.	48.	67.	64.	123.	64.	90.	8.	76.	38.	707.	185.
9	18.	17.	27.	30.	30.	34.	49.	22.	20.	2.	29.	21.	222.	104.

TABLE B-8 (Continued)

Requirements for 1991

IPAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	3709.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	809.	346.	371.	18792.	564.	932.	153.	9773.	1173.	213.	106.	93.	0.	0.
4	968.	6712.	854.	0.	4858.	2415.	1421.	8137.	4198.	381.	297.	134.	0.	0.
5	984.	6118.	647.	0.	3957.	2232.	1006.	6343.	3701.	580.	281.	162.	0.	75.
6	769.	4704.	740.	0.	3698.	2279.	732.	4328.	3055.	502.	193.	110.	219.	1347.
7	247.	1949.	429.	0.	1525.	1163.	460.	2623.	1385.	149.	103.	49.	131.	614.
8	67.	522.	76.	0.	520.	185.	107.	557.	289.	36.	40.	1.	25.	155.
9	35.	224.	25.	0.	235.	81.	35.	245.	145.	15.	9.	2.	12.	88.

IPAY	4700 ML	3700 NM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	34.	2317.	54.	343.	3845.	46.	0.	3195.	281.	183.	230.	94.	17.	938.
4	72.	9908.	214.	871.	5098.	246.	0.	4006.	390.	603.	519.	326.	41.	1590.
5	69.	6907.	158.	1147.	4606.	199.	0.	2962.	533.	195.	540.	375.	50.	2057.
6	59.	6540.	125.	732.	3662.	199.	1033.	2372.	510.	208.	388.	338.	59.	2089.
7	40.	2280.	60.	241.	1361.	94.	634.	705.	179.	68.	144.	114.	21.	927.
8	5.	952.	30.	49.	390.	37.	225.	356.	60.	17.	33.	28.	0.	245.
9	2.	468.	15.	28.	184.	18.	129.	87.	22.	18.	34.	4.	0.	172.

IPAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	5852.	0.	0.	0.	0.	0.	0.	0.	0.	13077.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	484.	569.	3456.	343.	46436.	1116.	1192.	714.	1501.	0.	508.	515.	2900.	154855.
4	593.	1280.	5326.	300.	0.	1608.	2535.	1126.	3319.	0.	1235.	706.	2900.	124826.
5	779.	1135.	5146.	292.	0.	1309.	2874.	932.	2320.	0.	980.	895.	3771.	116402.
6	519.	1041.	3283.	216.	0.	1039.	2296.	713.	2337.	0.	960.	614.	3275.	94571.
7	124.	767.	1499.	90.	0.	252.	1300.	246.	838.	0.	495.	201.	1495.	40230.
8	36.	142.	555.	25.	0.	115.	442.	50.	448.	0.	174.	78.	387.	12444.
9	21.	57.	158.	10.	0.	71.	172.	38.	99.	0.	64.	1.	118.	5306.

**ANNEX B-1**

**PROGRAM LISTING TO EXTRACT AND  
TABULATE ENLISTED MASTER RECORD INVENTORY**

LISTING OF COBOL PROGRAM  
TO EXTRACT AND TABULATE  
EMR INVENTORY DATA

```

IDENTIFICATION DIVISION.
PROGRAM-ID.          TABLE4.
AUTHOR.             KBGARVEY.
INSTALLATION.       Center for Naval Analyses.
DATE-WRITTEN.       OCTOBER 1986.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER.    VAX11-785(VMS).
OBJECT-COMPUTER.    VAX11-785(VMS).
SPECIAL-NAMES.     CO1 IS NEW-PAGE.
INPUT-OUTPUT        SECTION.
FILE-CONTROL.
    SELECT LABFILE ASSIGN TO      LFILE.
    SELECT INFILE ASSIGN TO       IFILE
                                FILE STATUS IS IN-STATUS.
    SELECT OUTFILE ASSIGN TO      OFILE
                                FILE STATUS IS OUT-STATUS.
    SELECT LISTING ASSIGN TO      LISTING.
DATA      DIVISION.
FILE      SECTION.
*
FD        LABFILE.
01        LABREC.
           03  LAB-TITLE          PIC XXX.
           03  LAB-MO             PIC X.
           03  LAB-YR             PIC 99.
           03  FILLER             PIC X(5).

FD        INFILE
RECORD CONTAINS 256 CHARACTERS
BLOCK CONTAINS 50 RECORDS
DATA RECORD IS SHORTEMR.
01  SHORTEMR.
     03  SSN                     PIC X(9).
     03  SCIND                   PIC X(5).
     03  STR-GAIN                 PIC 9(4).
     03  STR-LOSS                 PIC 9(4).
     03  DATE-OF-BIRTH           PIC 9(6).
     03  SEX                     PIC X.
     03  RACE                    PIC X.
     03  ETHNIC                  PIC X.
     03  HOME-OF-RECORD          PIC XX.
     03  PDEPS                   PIC X.
     03  PRESENT-RATE.
           05  PRATE-CODE         PIC 9(4).
           05  PAYGRADE          PIC 9.

```



03	TIME-IN-RATE.		
	05	TIME-IN-RATE-YY	PIC 99.
	05	TIME-IN-RATE-MM	PIC 99.
03	EFF-DATE-PAYGRADE.		
	05	EFF-DATE-YY	PIC 99.
	05	EFF-DATE-MM	PIC 99.
03	PNEC.		
	05	PNEC-CODE	PIC X(4).
	05	PNEC-DATE	PIC 9(4).
* ..... < 55 char >			
03	SNEC.		
	05	SNEC-CODE	PIC X(4).
	05	SNEC-DATE	PIC 9(4).
03	CURRENT-PRO-PAY.		
	05	PRO-SKILL	PIC X(4).
	05	PRO-LEVEL	PIC X.
	05	PRO-AUTH	PIC X.
03	VRB		PIC X(4).
03	RQC		PIC XX.
03	SOFT-EAOS.		
	05	SOFT-YY	PIC 99.
	05	SOFT-MM	PIC 99.
03	BRANCH-CLASS		PIC XX.
03	ADSD		PIC 9(6).
03	CADD.		
	05	CADD-YY	PIC 99.
	05	CADD-MM	PIC 99.
03	CED.		
	05	CED-YY	PIC 99.
	05	CED-MM	PIC 99.
03	PEBD.		
	05	PEBD-YY	PIC 99.
	05	PEBD-MM	PIC 99.
* ..... < 99 char >			
03	HARD-EAOS.		
	05	HARD-YY	PIC 99.
	05	HARD-MM	PIC 99.
03	EXTENSIONS.		
	05	SCHL-EXT	PIC XX.
	05	OTHER-EXT	PIC XX.
	05	OPEX	PIC XX.
03	ENLISTMENTS.		
	05	ENL-TYP	PIC XX.
	05	ENL-TERM	PIC 9.
	05	ENL-NUMBER	PIC 9.
03	TAR-SCORES-STAR		PIC X.
03	EDUCATION.		
	05	ED-YRS	PIC 99.
	05	ED-CERT	PIC X.
03	ONBOARD-ACTIVITY.		
	05	ACTUAL-UIC	PIC X(5).

05	SEA-SHORE-CODE	PIC X.	
05	ACTY-10DIGIT	PIC X(10).	
05	ACC	PIC XXX.	
05	DATE-RECD.		
	07 RECD-YY	PIC 99.	
	07 RECD-MM	PIC 99.	
05	PRD-DATE	PIC 9(4).	
05	PRD-REASON	PIC X(4).	
05	DNEC1	PIC X(4).	
* ..... < 152 char >			
	05 DNEC2	PIC X(4).	
03	PAST-ACTIVITY.		
	05 P-ACTUAL-UIC	PIC X(5).	
	05 P-SEA-SHORE-CODE	PIC X.	
	05 P-ACTY-10DIGIT	PIC X(10).	
	05 P-RATE-CODE	PIC 9(4).	
	05 P-PAYGRADE	PIC 9.	
	05 P-ACC	PIC XXX.	
	05 P-DATE-RECD.		
	07 P-RECD-YY	PIC 99.	
	07 P-RECD-MM	PIC 99.	
	05 P-DATE-TRANS.		
	07 P-TRANS-YY	PIC 99.	
	07 P-TRANS-MM	PIC 99.	
03	NAVY-LOSS-CODE	PIC X(3).	
03	DOD-LOSS-CODE	PIC X(3).	
03	DOD-AFEES	PIC X(3).	
03	CURRENT-EVAL.		
	05 OVERALL	PIC X.	
* ..... < 198 char >			
03	SCHOOL-HISTORY.		
	05 COMPLETION-DATE-1	PIC 9(4).	
	05 COURSE-CODE-1	PIC X(4).	
	05 STUD-ACTION-1	PIC XX.	
	05 COMPLETION-DATE-2	PIC 9(4).	
	05 COURSE-CODE-2	PIC X(4).	
	05 STUD-ACTION-2	PIC XX.	
	05 COMPLETION-DATE-3	PIC 9(4).	
	05 COURSE-CODE-3	PIC X(4).	
	05 STUD-ACTION-3	PIC XX.	
03	TYPE-ACQUISITION	PIC 99.	
* ..... < 230 char >			
03	SPECIAL-PROGRAM-CODES.		
	05 SPC-CODE	PIC X.	
	05 PROGENLFOR	PIC X.	
	05 SPC-REST	PIC XXX.	
03	RC-BRCL-FROM	PIC XX.	
03	SPEC-PROG-IND-1	PIC X.	
03	SPEC-PROG-YR	PIC X.	
03	SPEC-PROG-TAR	PIC X.	
03	AFQT	PIC 99.	
03	MG	PIC 9.	

```

03  SRB-ZONE-DATA.
    05  ZONE-ID                      PIC X.
    05  SRB-EFF-DATE                PIC 9(4).
    05  SRB-RT-SKILL-NEC            PIC X(4).
    05  SRB-PAY-CAT                 PIC X.
    05  SRB-FY-PAY                  PIC X.
    05  SRB-SKILL                   PIC X.
    05  SRB-LEVEL                   PIC X.
* ..... < 256 char. >

FD  OUTFILE.
01  OUTREC.
    05  FILLER                      PIC X(126).

FD  LISTING.
01  PRINTLINE                      PIC X(132).

WORKING-STORAGE SECTION.
*
01  INPARITY-FLAG                  PIC 9  VALUE 0.
01  OUTPARITY-FLAG                PIC 9  VALUE 0.
01  ERROR-FLAG                    PIC 9  VALUE 0.
01  NO-WRITE                      PIC 9  VALUE 0.
01  NO-LABEL                      PIC 9  VALUE 0.
01  FOUND                         PIC 9  VALUE 0.
01  END-OF-FILE                   PIC 9.
    88  EOF                       VALUE 1.
01  END-OF-LABEL                  PIC 9.
    88  EOL                       VALUE 1.
*
01  COUNTERS  USAGE IS COMP.
    03  RECIIN                    PIC 9(7).
    03  OUT-COUNT                 PIC 9(7).
    03  OTHER-COUNT              PIC 9(7).
    03  TARGET-COUNT             PIC 9(7).
    03  BAD-LOS                  PIC 9(7).
    03  BAD-RATING               PIC 9(7).
    03  BAD-PAYGRADE            PIC 9(7).
    03  BAD-PROGRAM              PIC 9(7).
    03  SPC-4YO                  PIC 9(7).
    03  SPC-6YO                  PIC 9(7).
    03  SPC-AM                   PIC 9(7).
    03  SPC-PS                   PIC 9(7).
    03  SPC-TAR                  PIC 9(7).
    03  SUBFARER                 PIC 9(7).
    03  RATING-5YO               PIC 9(7).
    03  XTRA-4YO                 PIC 9(7).
    03  BLANK-TAR                PIC 9(7).
    03  BLANK-AM                 PIC 9(7).
    03  OBLIG-4YO                PIC 9(7).

```

03	OBLIG-6YO	PIC 9(7).
03	PROGRAM-6	PIC 9(7).
03	PROGRAM-7	PIC 9(7).
03	SAVED-SEARCH	PIC 9(7).
03	NO-WRITE-LOS	PIC 9(7).
03	NO-WRITE-PG	PIC 9(7).
03	NO-WRITE-RATING	PIC 9(7).

\*\*\*\*\*  
 \* Rating Table Version 860707  
 \*

\* This is a COBOL table storing all enlisted ratings in the Navy.  
 \* It is designed to provide an easy method of associating each  
 \* 4-digit 'RATE-CODE' with its equivalent alphabetic  
 \* 'RATE-ABBREVIATION'.  
 \*

\* o FC's (Fire Controlman) added  
 \* o WT's (Weapons Technician) added  
 \*

\*\*\*\*\*

01 RATE-TABLE.

03	RATE-LIST.
05	FILLER PIC X(49)
VALUE	"7800AA 6700AB 6704ABE6705ABF6706ABH6600AC 6200AD "
05	FILLER PIC X(49)
VALUE	"6206ADJ6205ADR6800AE 6080AF 7100AG 7300AK 6900AM "
05	FILLER PIC X(49)
VALUE	"6903AME6902AMH6901AMS7800AN 6500AO 6520AQ 7800AR "
05	FILLER PIC X(49)
VALUE	"7500AS 7501ASE7502ASH7503ASM6300AT 6180AV 6400AW "
05	FILLER PIC X(49)
VALUE	"6310AX 7400AZ 0100BM 4020BR 4000BT 5600BU 6000CA "
05	FILLER PIC X(49)
VALUE	"5300CE 5500CM 6000CN 6000CR 1622CTA1666CTI1633CTM "
05	FILLER PIC X(49)
VALUE	"1644CTO1655CTR1611CTT5080CU 8300DA 2100DK 3200DM "
05	FILLER PIC X(49)
VALUE	"8300DN 1900DP 8300DR 1010DS 8300DT 5100EA 4100EM "
05	FILLER PIC X(49)
VALUE	"3800EN 5410EO 5380EQ 1000ET 1001ETN1002ETRO350EW "
05	FILLER PIC X(49)
VALUE	"5000FA 0700FC 5000FN 5000FR 0800FT 0803FTB0801FTG "
05	FILLER PIC X(49)
VALUE	"0802FTMO600GM 0604GMG0601GMM0602GMT4400GS 4401GSE "
05	FILLER PIC X(49)
VALUE	"4402GSM8000HA 8000HM 8000HN 8000HR 4300HT 4200IC "
05	FILLER PIC X(49)
VALUE	"1100IM 2300IS 2600JO 3100LI 1750LN 0150MA 4700ML "
05	FILLER PIC X(49)
VALUE	"3700MM 0900MN 3900MR 2200MS 0810MT 3300MU 1400NC "
05	FILLER PIC X(49)
VALUE	"1200OM 0300OS 0450OT 0451OTA0452OTM2700PC 7600PH "

05 FILLER PIC X(49)  
 VALUE "1080PI 4600PM 1800PN 7000PR 0200QM 1500RM 2500RP "  
 05 FILLER PIC X(49)  
 VALUE "3600SA 2490SH 2000SK 0250SM 3600SN 3600SR 0400ST "  
 05 FILLER PIC X(49)  
 VALUE "0401STG0404STS5700SW 7200TD 0500TM 05COTMS0500TMT"  
 05 FILLER PIC X(28)  
 VALUE "5800UT 0610WT 1700YN "

03 RATE-EACH REDEFINES RATE-LIST.

05 RATE-CODE-AND-ABBREV OCCURS 129 TIMES  
 ASCENDING KEY IS RATE-ABBREV  
 INDEXED BY R-INDEX.

07 RATE-CODE PIC 9(4).  
 07 RATE-ABBREV PIC XXX.

01 TARGET-TABLE USAGE IS COMP.

02 PROGRAM-LEVEL OCCURS 7 TIMES.

03 RT-LEVEL OCCURS 129 TIMES.

05 PG-LEVEL OCCURS 9 TIMES.

07 LOS-LEVEL OCCURS 21 TIMES.

09 ENTRY PIC 9(7).

\*

01 RATING-HOLD.

03 PRATE-HOLD PIC 9(4).

88 RT-5

VALUE 2600 7600 6600 1900 5100 5600 8300  
 5300 5700 5410 5500 5800 6000.

03 PREVIOUS-RATE-CODE

PIC 9(4).

01 TYPE-ACQUISITION-HOLD.

03 TYP-ACQ-HOLD

PIC 99.

88 ROMEO

VALUE 21 THRU 25 45 46 47.

88 USNR

VALUE 18 48.

88 TAR-TYPE

VALUE 19.

01 SPECIAL-PROGRAM-HOLD.

02 TAR-HOLD

PIC X.

88 TAR-PROGRAM VALUE "V".

01 PROGRAM-HOLD.

02 PROG-ENL-TYPE

PIC X.

02 PROG-ENL-FOR.

03 PEF

PIC X.

88 FOUR-YO

VALUE "H" "4" "F" "K" "E".

88 AM

VALUE "M" "Z" "W".

88 SIX-YO

VALUE "G" "5" "A" "B".

88 PRIOR-SERV

VALUE "S" "8" "7" "N" "Q" "L".

88 OTHER-TYPE

VALUE "\*" "1" "9" "C" "X" "D".

88 BLANKZ

VALUE " ".

88 TARS

VALUE "Y".

88 PSI

VALUE "J".

03 PEF-REST.

05 PEF1

PIC X.

05 PEF1-REST.

07 PEF2

PIC X.

07 PEF2-REST

PIC X.

```

01 SCIND-HOLD.
   03 SC1 PIC XX.
*      SCIND code begins with this or they're a LOSS
      88 GAIN VALUE "XF".
   03 FILLER PIC XXX.
01 BR-CL-AREA.
   03 BRCL-HOLD PIC XX.
      88 USN VALUE "11".
01 ACTY-HOLD.
   03 ACTY-TYPE PIC 9(4).
   03 HULL-NUMBER PIC 9(4).
   03 FILLER PIC XX.
*
01 WORK-AREA.
   03 ADSD-MONTHS PIC 9(6).
   03 NOW-MONTHS PIC 9(6).
   03 ADSD-HOLD.
      05 ADSD-Y PIC 99.
      05 ADSD-M PIC 99.
      05 ADSD-D PIC 99.
   03 SEAOS-MONTHS PIC S9(6).
   03 SEAOS-HOLD.
      05 SEAOS-Y PIC 99.
      05 SEAOS-M PIC 99.
   03 CED-MONTHS PIC S9(6).
   03 CED-HOLD.
      05 CED-Y PIC 99.
      05 CED-M PIC 99.
   03 OBLIG PIC S9(6).
01 EMR-DATE.
   03 EMR-YR PIC 99.
   03 EMR-MO PIC 99.
*      this assumes the EMR is created at the end of the month...
   03 EMR-DA PIC 99 VALUE 30.
*
01 IO-WORK-AREA.
   03 IN-STATUS.
      05 INSTAT PIC 9.
      05 INSTAT2 PIC 9.
   03 OUT-STATUS.
      05 OUTSTAT PIC 9.
      05 OUTSTAT2 PIC 9.
*
01 PAR-COUNT-TABLE USAGE IS COMP.
   03 PARCOUNTS OCCURS 10 TIMES.
      05 INPARITY-COUNT PIC 9(7).
      05 OUTPARITY-COUNT PIC 9(7).

```

```

01 SUBSCRIPTS      USAGE IS COMP.
    03 P           PIC S999.
    03 S           PIC S999.
    03 R           PIC S999.
    03 I           PIC S999.
    03 L           PIC S999.
    03 LOS         PIC S999.
    03 PG          PIC S999.
    03 PM          PIC S999.
    03 PREVIOUS-R  PIC S999.

01 PROGRAM-LABELS.
    03 PROGRAM-LABEL-ALL.
        05 FILLER PIC X(20) VALUE "4-year Obligors" ".
        05 FILLER PIC X(20) VALUE "Active Mariners" ".
        05 FILLER PIC X(20) VALUE "5 & 6 year Obligors" ".
        05 FILLER PIC X(20) VALUE "Prior Service" ".
        05 FILLER PIC X(20) VALUE "TARs" ".
        05 FILLER PIC X(20) VALUE "LOS 10 & Greater" ".
        05 FILLER PIC X(20) VALUE "Other/No Program" ".
    03 PROGRAM-LABEL-EACH REDEFINES PROGRAM-LABEL-ALL.
        05 PLABEL OCCURS 7 TIMES PIC X(20).

01 TITLE-LINE.
    05 FILLER PIC X(60) VALUE SPACES.
    05 TITLE-SLOT PIC XXX.

01 DETAIL-LINE.
    03 SLOT OCCURS 21 TIMES PIC ZZZZZ9.
*

01 REC-LINE.
    05 PART-1 PIC X(112).
    05 PART-2 PIC X(105).
    05 PART-3 PIC X(58).
*

01 M-LINE.
    05 PROSE PIC X(30).
    05 M-NUM PIC ZZZZZZZZZ9.

01 PARITY-REASON-LIST.
    03 PARITY-LIST.
        05 FILLER PIC X(20) VALUE " AT END INVLD. KEY".
        05 FILLER PIC X(20) VALUE "PERM ERROR*****".
        05 FILLER PIC X(20) VALUE "*****".
        05 FILLER PIC X(20) VALUE "*****".
        05 FILLER PIC X(20) VALUE "DEC ERROR *****".
    03 PARITY-EACH REDEFINES PARITY-LIST.
        05 PARITY-REASON OCCURS 10 TIMES PIC X(10).
*

01 DISPLAY-WINDOW PIC ZZZZZ99V99.
01 REC-WINDOW PIC -----9.

```

PROCEDURE DIVISION.

```
*****
*      This program processes the SHORT EMR.
*      It is set up to create and write a 4-dimensional table,
*      1. Entry program (7)
*      2. Rating (129)
*      3. Paygrade (9)
*      4. Length of Service in years (21)
*****
*
```

DECLARATIVES.

BAD-INPUT SECTION.

```
      USE AFTER STANDARD ERROR          PROCEDURE ON INFILE.
BAD-INPUT-PARA.
      IF INSTAT > 0
        ADD 1 TO INPARITY-COUNT (INSTAT)
        IF IN-STATUS > "29"
          DISPLAY "INPUT STATUS IS " IN-STATUS
          MOVE 1 TO INPARITY-FLAG.
```

END DECLARATIVES.

\*

MAIN-CODING SECTION.

LABEL-PARA.

```
      OPEN INPUT LABFILE.
      READ LABFILE AT END MOVE 1 TO NO-LABEL.
      MOVE LAB-YR TO EMR-YR.
      IF LAB-MO = "M"
        MOVE 03 TO EMR-MO
      ELSE IF LAB-MO = "J"
        MOVE 06 TO EMR-MO
      ELSE IF LAB-MO = "S"
        MOVE 09 TO EMR-MO
      ELSE IF LAB-MO = "D"
        MOVE 12 TO EMR-MO
      ELSE MOVE "999999" TO EMR-DATE
        MOVE 1 TO NO-LABEL.
      DISPLAY "EMR LABEL IS " EMR-DATE.
      CLOSE LABFILE.
```

```
      COMPUTE NOW-MONTHS = (EMR-YR * 12) + EMR-MO.
INITIAL-PARA.
```

```
      OPEN INPUT INFILE,
      OUTPUT OUTFILE, LISTING.
```

```
      INITIALIZE COUNTERS, PAR-COUNT-TABLE.
```

\* Echo the date read in as the "current" date...

```
      DISPLAY "  NOW set for ", LAB-MO," 19",LAB-YR," for LOS purposes...".
```

```
      MOVE 0 TO END-OF-FILE.
```

```
      READ INFILE AT END MOVE 1 TO END-OF-FILE.
```



PERFORM READ-WRITE-CYCLE  
UNTIL EOF.  
PERFORM PROGRAM-TABLE-DUMP VARYING PM FROM 1 BY 1  
UNTIL PM > 7.  
SECOND-PARA.  
PERFORM STATS-ROUTINE.  
FINISH-UP.  
CLOSE LISTING.  
CLOSE OUTFILE.  
CLOSE INFILE.  
STOP RUN.

\*  
\*\*\*\*\* SUBROUTINE PARAGRAPHS \*\*\*\*\*

READ-WRITE-CYCLE.

MOVE SCIND TO SCIND-HOLD.

\* limit this to only Active Strength Gains

IF GAIN

ADD 1 TO TARGET-COUNT

PERFORM REST-OF-PROGRAM.

READ INFILE AT END MOVE 1 TO END-OF-FILE.

ADD 1 TO RECIN.

REST-OF-PROGRAM.

MOVE 0 TO NO-WRITE.

\*\* LOS (length of service) IN MONTHS

IF ADSD IS NUMERIC AND (ADSD > "000000")

PERFORM LOS-COMPUTE

ELSE

ADD 1 TO NO-WRITE-LOS

MOVE 1 TO NO-WRITE.

IF NO-WRITE = 0

IF LOS > 9

ADD 1 TO PROGRAM-6

MOVE 6 TO PM

IF LOS > 20

MOVE 21 TO LOS

PERFORM CONTINUE-WITH-PAYGRADE

ELSE

PERFORM CONTINUE-WITH-PAYGRADE

ELSE

PERFORM DETERMINE-PROGRAM

PERFORM CONTINUE-WITH-PAYGRADE.

DETERMINE-PROGRAM.

\*\*\*\*\*

\* SPECIAL-PROGRAM field on the EMR is used to determine program  
\* of entry. If this information is missing or inconclusive, the  
\* procedures MEASURE-THE-OBLIGATION and CHECK-BLANKS-FOR-PROGRAM  
\* attempt to use other means to determine program of entry.

\*\*\*\*\*

MOVE SPECIAL-PROGRAM-CODES TO PROGRAM-HOLD.

MOVE PRATE-CODE TO PRATE-HOLD.

MOVE SPEC-PROG-TAR TO TAR-HOLD.

```

      IF TARS
          OR TAR-PROGRAM
          ADD 1 TO SPC-TAR
          MOVE 5 TO PM
      ELSE IF FOUR-YO
          ADD 1 TO SPC-4YO
          MOVE 1 TO PM
      ELSE IF AM
          IF PEF1 = "4"
              ADD 1 TO SUBFARER
              MOVE 1 TO PM
          ELSE
              ADD 1 TO SPC-AM
              MOVE 2 TO PM
      ELSE IF SIX-YO
          ADD 1 TO SPC-6YO
          MOVE 3 TO PM
      ELSE IF PRIOR-SERV
          ADD 1 TO SPC-PS
          MOVE 4 TO PM
      ELSE IF RT-5
          ADD 1 TO RATING-5YO
          MOVE 3 TO PM
      ELSE IF PSI OR OTHER-TYPE
          IF ENL-NUMBER = 1
              PERFORM MEASURE-THE-OBLIGATION
          ELSE
              MOVE 1 TO PM
              ADD 1 TO XTRA-4YO
      ELSE IF BLANKZ
          PERFORM CHECK-BLANKS-FOR-PROGRAM
      ELSE
          ADD 1 TO PROGRAM-7
          MOVE 7 TO PM.

```

CHECK-BLANKS-FOR-PROGRAM.

\* use TYPE-OF\_ACQUISITION to determine program of entry...

```

      MOVE TYPE-ACQUISITION TO TYP-ACQ-HOLD.
      IF TAR-TYPE
          ADD 1 TO BLANK-TAR
          MOVE 5 TO PM
      ELSE IF USNR
          ADD 1 TO BLANK-AM
          MOVE 2 TO PM
      ELSE IF ENL-NUMBER = 1
          PERFORM MEASURE-THE-OBLIGATION
      ELSE
          ADD 1 TO PROGRAM-7
          MOVE 7 TO PM.

```

MEASURE-THE-OBLIGATION.

\*\*\*\*\*  
 \* Use current obligation (1st termers only) to determine program of entry...  
 \* Those with 4 year obligations go to 4YO program, those with greater  
 \* go to the 5/6YO program...  
 \*\*\*\*\*

MOVE SOFT-EAOS TO SEAOS-HOLD.  
 MOVE CED TO CED-HOLD.  
 COMPUTE CED-MONTHS ROUNDED = CED-M + (CED-Y \* 12).  
 COMPUTE SEAOS-MONTHS ROUNDED = SEAOS-M + (SEAOS-Y \* 12).

SUBTRACT CED-MONTHS FROM SEAOS-MONTHS GIVING OBLIG ROUNDED.

IF OBLIG > 48  
     ADD 1 TO OBLIG-6YO  
     MOVE 3 TO PM  
 ELSE  
     ADD 1 TO OBLIG-4YO  
     MOVE 1 TO PM.

CONTINUE-WITH-PAYGRADE.

IF PAYGRADE IS NUMERIC  
     MOVE PAYGRADE TO PG  
     IF PG > 0 AND (PG < 10)  
         PERFORM CONTINUE-WITH-RATING  
 ELSE  
     ADD 1 TO NO-WRITE-PG  
     MOVE 1 TO NO-WRITE  
 ELSE  
     ADD 1 TO NO-WRITE-PG  
     MOVE 1 TO NO-WRITE.

CONTINUE-WITH-RATING.

IF PRATE-CODE = PREVIOUS-RATE-CODE  
     ADD 1 TO SAVED-SEARCH  
     MOVE PREVIOUS-R TO R  
     MOVE 1 TO FOUND  
 ELSE  
     MOVE 0 TO FOUND  
     PERFORM TABLE-SEARCH VARYING I FROM 1 BY 1  
         UNTIL FOUND = 1 OR (I > 129).  
 IF FOUND = 0  
     ADD 1 TO NO-WRITE-RATING  
     MOVE 1 TO NO-WRITE  
 ELSE  
     IF NO-WRITE = 0  
         ADD 1 TO ENTRY (PM,R,PG,LOS).

\*\*\*\*\* END OF REST-OF-PROGRAM \*\*\*\*\*

TABLE-SEARCH.

IF RATE-CODE (I) = PRATE-CODE  
 MOVE PRATE-CODE TO PREVIOUS-RATE-CODE  
 MOVE I TO R, PREVIOUS-R  
 MOVE 1 TO FOUND.

LOS-COMPUTE.

MOVE ADSD TO ADSD-HOLD.  
 COMPUTE ADSD-MONTHS = (ADSD-Y \* 12) + ADSD-M.  
  
 COMPUTE L = NOW-MONTHS - (ADSD-MONTHS).  
 COMPUTE LOS = (L / 12) + 1.

PROGRAM-TABLE-DUMP.

PERFORM RATE-TABLE-DUMP VARYING R FROM 1 BY 1  
 UNTIL R > 129.

RATE-TABLE-DUMP.

PERFORM PAYGRADE-TABLE-DUMP VARYING PG FROM 1 BY 1  
 UNTIL PG > 9.

PAYGRADE-TABLE-DUMP.

PERFORM LOS-DUMP VARYING LOS FROM 1 BY 1  
 UNTIL LOS > 21.  
 WRITE OUTREC FROM DETAIL-LINE AFTER 1 LINE.

LOS-DUMP.

MOVE ENTRY (PM,R,PG,LOS) TO SLOT (LOS).

WRITE-LINE.

MOVE SHORTEMR TO REC-LINE.  
 WRITE PRINTLINE FROM PART-1 AFTER ADVANCING 3 LINES.  
 WRITE PRINTLINE FROM PART-2 AFTER ADVANCING 1 LINE.  
 WRITE PRINTLINE FROM PART-3 AFTER ADVANCING 1 LINE.  
 MOVE SPACES TO PRINTLINE.  
 WRITE PRINTLINE AFTER ADVANCING 1 LINE.

\*

STATS-ROUTINE.

MOVE "INPUT RECS " TO PROSE.  
 MOVE RECIN TO M-NUM.  
 WRITE PRINTLINE FROM M-LINE AFTER 2 LINES.  
 MOVE "PARITY ERRORS" TO PRINTLINE.  
 WRITE PRINTLINE AFTER 2 LINES.  
 PERFORM PARITY-INFO-DUMP VARYING I FROM 1 BY 1  
 UNTIL I > 9.  
 MOVE "NO. OF ACTIVE" TO PROSE.  
 MOVE TARGET-COUNT TO M-NUM.  
 WRITE PRINTLINE FROM M-LINE AFTER 2 LINES.  
  
 MOVE "No. of Saved Rate Searches" TO PROSE.  
 MOVE SAVED-SEARCH TO M-NUM.  
 WRITE PRINTLINE FROM M-LINE AFTER 2 LINES.

```

MOVE "OUTPUT RECORDS" TO PROSE.
MOVE OUT-COUNT TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 2 LINES.

*

MOVE "Program Enlisted For 4 YO" TO PROSE.
MOVE SPC-4YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 3 LINES.
MOVE "        6 YO" TO PROSE.
MOVE SPC-6YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "        A.M." TO PROSE.
MOVE SPC-AM TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "        P.S." TO PROSE.
MOVE SPC-PS TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "        TAR " TO PROSE.
MOVE SPC-TAR TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "        4 YO Subfarer " TO PROSE.
MOVE SUBFARER TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE " 5 YO Rating " TO PROSE.
MOVE RATING-5YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE " PSI or Other 4 YO " TO PROSE.
MOVE XTRA-4YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "TAR by TYPE-ACQUISITION " TO PROSE.
MOVE BLANK-TAR TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "AM by TYPE-ACQUISITION " TO PROSE.
MOVE BLANK-AM TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "Obligation 4 YO " TO PROSE.
MOVE OBLIG-4YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "Obligation 6 YO " TO PROSE.
MOVE OBLIG-6YO TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "LOS Greater than 9 " TO PROSE.
MOVE PROGRAM-6 TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "Unknown Program (7) " TO PROSE.
MOVE PROGRAM-7 TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "Not Written LOS " TO PROSE.
MOVE NO-WRITE-LOS TO M-NUM.
WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
MOVE "Not Written Paygrade " TO PROSE.

```

```

        MOVE NO-WRITE-PG          TO M-NUM.
        WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
        MOVE "Not Written Rating" TO PROSE.
        MOVE NO-WRITE-RATING      TO M-NUM.
        WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.
PARITY-INFO-DUMP.
        IF INPARITY-COUNT (I) > 0 OR (OUTPARITY-COUNT (I) > 0)
            MOVE PARITY-REASON (I) TO PROSE
            MOVE INPARITY-COUNT (I) TO M-NUM
            WRITE PRINTLINE FROM M-LINE AFTER 1 LINE
            MOVE "          OUTPUT" TO PROSE
            MOVE OUTPARITY-COUNT (I) TO M-NUM
            WRITE PRINTLINE FROM M-LINE AFTER 1 LINE.

        IF I = 3
            MOVE 8 TO I.
*
PREMATURE-ABORT.
        MOVE "<<< EXCESSIVE PARITY ERRORS ENCOUNTERED >>>"
            TO PRINTLINE.
        WRITE PRINTLINE AFTER 5 LINES.
        MOVE "<<< RUN IS TERMINATING PREMATURELY >>>" TO
            PRINTLINE.
        WRITE PRINTLINE AFTER 2 LINES.
        GO TO SECOND-PARA.

```

ANNEX B-2

APL PROGRAM LISTINGS FOR COMPUTING  
AGGREGATE TRANSITION AND ADDITION RATES



APL PROGRAM LISTINGS  
FOR COMPUTING TRANSITION  
AND ADDITION RATES

FUNCTION: XTRANS\_TO

```

▽T←YEAR1 XTRANS_TO YEAR2
[1]  A THIS FUNCTION CONSTRUCTS THE TRANSITION RATES FROM
[2]  A YEAR1 TO YEAR2.
[3]  A
[4]  A INVENTORIES ARE DIVIDED INTO 5 ENTRY PROGRAMS
[5]  A ONLY FOR LOS 1 TO LOS 9. THEREFORE, TRANSITION RATES
[6]  A ARE COMPUTED SEPARATELY FOR THESE 2 PARTS.
[7]  A INVENTORIES OF LOS 1-9 ARE STORED IN "YMAT1" AND YMAT2".
[8]  A HIGHER LOS PARTS ARE STORED IN "SENIOR", "SENIOR1"
[9]  A "SENIOR2"...
[10] A
[11] A FILE 17 = THE FULL EMR INVENTORY
[12] A 19 = REDUCED (5 X 69 X 9 X 9) INVENTORY (LOS1 IS Y-1)
[13] A 13 = NEW FY PRIOR SERVICE ACCESSIONS
[14] A 8 = INVENTORY OF HIGH LOS (21-31) (SEPARATE FOR
[15] A STORAGE REASONS)
[16] A
[17] □DAS □CHANS
[18] CF←□ASS '17 CNA9: [GARVEYK.APL] SEPTEMBER/DA' A OPEN 17
[19] A
[20] CF←□ASS '19 CNA9: [GARVEYK.APL] SEPT/DA' A OPEN 19
[21] YMAT1←+/ [3] □ [88-Y1] 19
[22] YMAT2←+/ [3] □ [88-Y2] 19
[23] □DAS 19
[24] A
[25] A PRIOR SERVICE
[26] A FORM "NEWPS" WHICH IS THE NEW PRIOR SERVICE ACCESSIONS
[27] A IN THE END YEAR, SUBTRACT IT FROM THE END YEAR INVENTORY...
[28] CF←□ASS '13 CNA9: [GARVEYK.APL] PRIORSERVICE/DA'
[29] PRIOR_END←+/ [2] □ [88-Y2] 13
[30] PS←YMAT2 [4; ;]
[31] NEWPS←CONDENSE 129 9+PRIOR_END
[32] LESS_THAN←((,PS)<,NEWPS)/1ρ,PS
[33] +(O=ρLESS_THAN)/L1
[34] ' NEW BLOOD CELL EXCEEDS PRIOR SERVICE CELL. '
[35] NEWPS←,NEWPS
[36] NEWPS [LESS_THAN] +(,PS) [LESS_THAN]
[37] NEWPS←(MSHAPE,9)ρNEWPS
[38] L1:
[39] A REMOVE END-YEAR NEW PRIOR SERVICE ACCESSIONS FROM NUMERATOR
[40] YMAT2 [4; ;] ←PS - NEWPS
[41] A

```

```

[42]  A .....
[43]  A "LOS1_T" IS THE TOTAL (BY PROGRAM) OF LOS 1.
[44]  A IT WILL BE RE-SHAPED AND USED AS DENOMINATOR FOR
[45]  A FOR THE LOS 1 DIMENSION OF THE ADDITION MATRIX...
[46]  LOS1_T+T_ALLOCATION (88-Y1)
[47]  A .....
[48]  A
[49]  A FOR STORAGE REASONS, THE UPPER LOS INVENTORY (LOS 21-31)
[50]  A ARE IN FILE 8...
[51]  CF+ASS '8 CNA9: [GARVEYK.APL] HIGHLOS/DA' A OPEN 8
[52]  HIGH1+ +/ [2]  $\oplus$  [88-Y1] 8
[53]  HIGH2+ +/ [2]  $\oplus$  [88-Y2] 8
[54]  DAS 8
[55]  A
[56]  A "SENIOR" IS THE FULL BEGIN-YEAR INVENTORY...
[57]  A "SENIOR1" IS THE INVENTORY LOS 10-21 (THE PART OF
[58]  A THE INVENTORY NOT DIVIDED INTO ENTRY PROGRAMS).
[59]  SENIOR1+1 129 -12+5 0 0+SENIOR+ +/ [3]  $\oplus$  [88-Y1] 17
[60]  LOS9+CONDENSE +SENIOR [; ; 9]
[61]  SENIOR1+(+1 129 11+SENIOR1), [2] HIGH1
[62]  SENIOR1+CONDENSE SENIOR1
[63]  SENIOR1+((( $\rho$ LOS9), 1) $\rho$ LOS9), [2] SENIOR1
[64]  A "SENIOR1" IS NOW LOS 9 - 31...
[65]  SENIOR2+1 129 -12+5 0 0+ +/ [3]  $\oplus$  [88-Y2] 17
[66]  A
[67]  A PRIOR SERVICE
[68]  A (SAME PROCEDURE THAT WAS PERFORMED ON LOWER LOS, NOW DONE
[69]  A ON THE HIGHER LOS END-YEAR INVENTORY...)
[70]  LESS_THAN+((, SENIOR2)<, NEWPS)/1 $\rho$ , NEWPS
[71]  + (0- $\rho$ LESS_THAN)/L2
[72]  ' NEW BLOOD CELL EXCEEDS PRIOR SERVICE CELL. '
[73]  NEWPS+ , NEWPS
[74]  NEWPS [LESS_THAN] + (, SENIOR2) [LESS_THAN]
[75]  L2:
[76]  NEWPS+1 129 12 $\rho$ NEWPS
[77]  SENIOR2+SENIOR2 - NEWPS A REMOVE PRIOR SERVICE
[78]  DAS 17
[79]  A
[80]  A "TRANS_DIVIDE_BY" DIVIDES END-YEAR INVENTORY BY BEGIN-YEAR,
[81]  A AND RETURNS A RESULT THAT CONTAINS BOTH THE TRANSITION
[82]  A AND ADDITION MATRIX. THE ADDITION MATRIX IS THE MATRIX
[83]  A OF EVERY CELL WHERE END-YEAR INVENTORY EXCEEDS BEGIN-YEAR.
[84]  SENIOR2+(+1 129 11+SENIOR2), [2] HIGH2
[85]  SENIOR2+CONDENSE SENIOR2
[86]  T1+SENIOR2 TRANS_DIVIDE_BY (0 -1+SENIOR1)
[87]  T2+T1 [2; ;] A "T2" IS THE ADDITION MATRIX (HIGH LOS)
[88]  T1+T1 [1; ;] A "T1" IS THE TRANSITION MATRIX (HIGH LOS)
[89]  A
[90]  YMAT1+(5, MSHAPE, 7)+((5, MSHAPE, 1)+YMAT1), [3] 0 0 2+YMAT1
[91]  YMAT2+(5, MSHAPE, 7)+ 2 $\phi$  YMAT2
[92]  A "YMAT1" BECOMES LOS 1 3 4 5 6 7 8
[93]  A "YMAT2" BECOMES LOS 3 4 5 6 7 8 9

```

```

[94] A=====
[95] A "TRANS_DIVIDE_BY" IS AGAIN USED TO GET TRANSITION/ADDITION
[96] A MATRICES FOR THE LOWER (1-9) LOS CELLS...
[97] A=====
[98] T3+YMAT2 TRANS_DIVIDE_BY YMAT1
[99] T4+T3 [2; ; ;]
[100] T3+(T3 [1; ; ;] ), [3] (5,MSHAPE,22) pT1 A COMBINE LOW/HIGH TRANSITION
[101] T4+T4, [3] (5,MSHAPE,22) pT2 A LOW/HIGH ADDITION
[102] A
[103] SENIOR+6 129++/SENIOR A SENIOR (THE BEGIN YEAR INVENTORY )
[104] SENIOR+CONDENSE SENIOR A IS SUMMED ACROSS PROG, PG AND LOS
[105] SENIOR++/SENIOR
[106] SENIOR+Q(5,MSHAPE) p540p, SENIOR
[107] SENIOR+Q(28,MSHAPE,5) pSENIOR A BEING PREPARED AS THE DENOMINATOR
[108] DENOM+LOS1_T, [3] SENIOR A TO COMPUTE THE "ADDITION" PERCENTS..
[109] T2+T3, [1] T4+T4 ZERO_DIVIDE_BY DENOM
[110] A
[111] A NEW RATINGS FC AND WT HAVE NO TRANSITION HISTORY...
[112] A WE ARE SUBSTITUTING THE RATING WHERE THEY CAME FROM...
[113] FC+MSHAPE A_TO_N 'FC '
[114] FTG+MSHAPE A_TO_N 'FTG'
[115] +(0=pFC)/NO_FC
[116] T2 [5+15;FC;] +0 A ADDERS FOR THE NEW RATINGS SUPPRESSED...
[117] T2 [15;FC;] +T2 [15;FTG;] A FC IS GIVEN THE SAME TRANSITION AS FTG
[118] NO_FC:
[119] WT+MSHAPE A_TO_N 'WT '
[120] +(0=pWT)/NO_WT
[121] GMT+MSHAPE A_TO_N 'GM '
[122] T2 [5+15;WT;] +0 A ADDERS FOR THE NEW RATINGS ARE SUPPRESSED
[123] T2 [15;WT;] +T2 [15;GMT;] A WT IS GIVEN THE SAME TRANSITION AS GMT
[124] NO_WT:
[125] DAS CHANS
[126] T+T2

```

▽

# FUNCTION: CONDENSE

```

▽MOUT+CONDENSE MIN
[1]  A THIS FUNCTION ACCEPTS A MATRIX AS INPUT THAT HAS ONE (AND ONLY
[2]  A ONE) DIMENSION EQUAL TO 129. THAT DIMENSION IS ASSUMED TO REPRESENT
[3]  A RATING, AND IS REDUCED FROM 129 TO "MSHAPE" (69 IN OUR APPLICATION)
[4]  A BY USING THE GLOBAL VARIABLES "CRATES" (INDEX OF CHIEF RATINGS),
[5]  A "FRATES" (INDEX OF "FEEDER" RATES) AND "FEED_VEC", WHICH GUIDES HOW
[6]  A FEEDER RATINGS ARE COLLAPSED INTO CHIEF RATINGS.
[7]  A
[8]  A   CRATES =  2  9 12 18 23 39 46 49 54 58 62 84 89 100 106
[9]  A
[10] A   FRATES =  3 4 5 7 8 13 14 15 12 17 19 20 21 22 25 29 31 32 45 48 53
[11] A           55 56 57 59 60 61 63 64 67 68 79 82 85 86 101 102 103 107
[12] A
[13] A   FEED_VEC =  1 1 1 2 5  3  3  3  2  5  4  4  4  5  5  6 15  8  6  8  9
[14] A           9  9  9 10 10 10 11 11  7 13  9 13 12 12  14  14  6 10
[15] A
[16] R←(ρMIN) 1 129
[17] SH←ρρMIN
[18] →(R>SH)/0
[19] A CHECKS FOR 1 DIMENSION OF SHAPE 129...
[20] NEWSHAPE←ρMIN
[21] NEWSHAPE[R] ←1
[22] A "ZERO_LINES" ARE EITHER OBSOLETE RATINGS OR REDUNDANT ONES
[23] A (IE. "SR" AND "SN" ARE REDUNDANT BECAUSE "SA" IS USED...)
[24] ZERO_LINES←8 9 18 21 32 38 39 50 52 54 61 62 66 67 80 81 82
[25] ZERO_LINES←ZERO_LINES,117 118 125 126
[26] A
[27] ZERO_IN←7 7 1 1 33 35 35 47 47 47 60 60 64 64 79 79 79 113
[28] ZERO_IN←ZERO_IN,113 124 124
[29] BOOL1←129ρ1
[30] BOOL1[ZERO_LINES] ←0
[31] C1←(R-1)ρ';'
[32] C2←(SH-R)ρ';'
[33] A
[34] K←1
[35] L1:
[36] COMAND←'MIN [' ,C1,'ZERO_IN[K] ' ,C2,'] +MIN [' ,C1,'ZERO_IN[K] ' ,C2
[37] COMAND←COMAND,'] + MIN [' ,C1,'ZERO_LINES[K] ' ,C2,'] '
[38] ▲COMAND
[39] K←K+1
[40] →(K≤(ρZERO_IN))/L1
[41] A
[42] COMAND←'MIN [' ,C1,'ZERO_LINES',C2,'] ←0'
[43] ▲COMAND
[44] COMAND←'M2←MIN [' ,C1,'(BOOL1/1129)',C2,'] '
[45] ▲COMAND
[46] A PUT ADJ AND ADJ FROM THE 129 LIST INTO AD OF THE 108 LIST...
[47] AD108←ATON 'AD '
[48] AD129←ATON129 'ADJADR'
[49] COMAND←'M2 [' ,C1,'AD108',C2,'] ←M2 [' ,C1,'AD108',C2,'] + NEWSHAPEρ+ / [R] MIN [
[50] COMAND←COMAND,C1,'AD129',C2,'] '
[51] ▲COMAND

```

```

[52]  * SAME FOR ETR ETN TO ET...
[53]  ET108←ATON 'ET '
[54]  ET129←ATON129 'ETRETN'
[55]  COMAND←'M2 [' ,C1,'ET108',C2,'] +M2 [' ,C1,'ET108',C2,'] + NEWSHAPEρ+ / [R] MIN ['
[56]  COMAND←COMAND,C1,'ET129',C2,'] '
[57]  *COMAND
[58]  J+1
[59]  TOP:
[60]  PATTERN_1←PATTERN_2←' [' ,((R-1)ρ';')
[61]  PATTERN_1←PATTERN_1,( *FRATES [J] ),((SH-R)ρ';'),'] '
[62]  PATTERN_2←PATTERN_2,( *CRATES [FEED_VEC [J] ] ),((SH-R)ρ';'),'] '
[63]  COMAND←'M2',PATTERN_2,'+M2',PATTERN_2,'+M2',PATTERN_1
[64]  *COMAND
[65]  * NEXT LINE DISPLAYS THE FEEDER-TO-CHIEF PROCESS...
[66]  * LIST108 [FRATES [J] ;5 6 7], ' TO ',LIST108 [CRATES [FEED_VEC [J] ] ;5 6 7]
[67]  * BUT IS COMMENTED OUT...
[68]  J+J+1
[69]  ←(J≤ρFRATES)/TOP
[70]  BOOL1←108ρ1
[71]  BOOL1 [FRATES] ←0
[72]  PATTERN_1←' [' ,((R-1)ρ';'),( * (BOOL1/1 108) ),((SH-R)ρ';'),'] '
[73]  *'MOUT←M2',PATTERN_1

```

#### FUNCTION: TRANS\_DIVIDE\_BY

```

▽Z ← X TRANS_DIVIDE_BY Y ;ZIP;SHAPE;N;NUM;DENOM;TOO_BIG;DIFF
[1]  * THE RULE HERE IS :
[2]  * NUM ÷ DENOM IS CONSTRAINED TO ( ≤ 1 )
[3]  * THE OUTPUT OF THIS FUNCTION IS THE MATRIX OF QUOTIENTS
[4]  * OBTAINED FROM X ÷ Y, COMBINED WITH AN IDENTICALLY-SHAPED
[5]  * "EXCESS" MATRIX, WHICH CONTAINS ZEROES EXCEPT WHERE X : Y.
[6]  * THOSE CELLS OF THE "EXCESS" MATRIX CONTAIN (X - Y).
[7]  *
[8]  SHAPE←ρY
[9]  N←1 * /SHAPE
[10]  DENOM←,Y
[11]  NUM←,X
[12]  TOO_BIG←(NUM>DENOM)/N
[13]  DIFF←(X>Y)*SHAPEρNUM-DENOM
[14]  ZIP←(NUM=0)/N
[15]  NUM [TOO_BIG] ←DENOM [TOO_BIG]
[16]  DENOM [ZIP] ←1
[17]  *
[18]  * IF X IS OF SHAPE (I,J), THEN THE OUTPUT Z IS SHAPE (2,I,J).
[19]  * Z [1;:] = TRANSITION MATRIX
[20]  * Z [2;:] = "EXCESS" MATRIX TO BE USED TO MAKE ADDITION MATRIX
[21]  *
[22]  Z←(2,SHAPE)ρ(SHAPEρNUM+DENOM), [1] DIFF

```

# FUNCTION: T\_ALLOCATION

```

▽MOUT ← T_ALLOCATION YEAR ;M;PRIOR;P;ALLOCATION
[1]  A THIS FUNCTION PROVIDES A TOTAL LOS 1 FOR "YEAR" TO THE FUNCTION
[2]  A "XTRANS_TO".
[3]  A
[4]  M←YMAT1
[5]  →(√/(5,MSHAPE,9)≠PM)/0
[6]  ALLOCATION←+/M[;;1]
[7]  A
[8]  A =====
[9]  A THE FOLLOWING IF "NEW-BLOOD" PRIOR SERVICE" ARE TO BE INCLUDED
[10] A AS LOS1 PRIOR SERVICE IN THE ALLOCATION MATRIX...
[11] A =====
[12] A
[13] PRIOR←+/ [2] ⌊ [(YEAR+1)] 13
[14] →(√/(129 21)≠PPRIOR)/0
[15] P←+/PRIOR+CONDENSE PRIOR
[16] ALLOCATION [4] ←+/P
[17] A
[18] MOUT←+/ALLOCATION
▽

```

# FUNCTION: A\_TO\_N

```

▽N ← SIZE A_TO_N RATING ;SH:A
[1]  A THIS FUNCTION IS A GENERAL PURPOSE METHOD OF PRODUCING INDECES
[2]  A INTO A LIST OF RATINGS, USING THE RATING ABBREVIATION
[3]  A
[4]  A GIVEN A GLOBAL VARIABLE "LIST∈∈", WHERE ∈∈ IS THE NUMBER OF
[5]  A ENTRIES AND THE VARIABLE IS OF SHAPE (∈∈,7) OF THE FORM:
[6]  A
[7]  A      1          0100BM_          ("_" DENOTES BLANK)
[8]  A      2          0150MA_
[9]  A      3          0401STG
[10] A
[11] A      .
[12] A      ∈∈          8300DN_
[13] A
[14] A THEN THE COMMAND " ∈∈ A_TO_N 'BM MA STGDN' "
[15] A WILL RETURN THE RESULT: 1 2 3 ∈∈
[16] A
[17] SH←OpO
[18] TOP:
[19] A←3+RATING
[20] A'SH←SH,(A^.=Q(',(√SIZE),',3)+4ΦLIST',(√SIZE),')/1',(√SIZE)
[21] RATING←3+RATING
[22] →((pRATING)>0)/TOP
[23] XIT:N←SH
▽

```

# FUNCTION: BUILD\_LOS1

```

▽MOUT ← BUILD_LOS1 YEAR ;M;P;PRIDE_PART;LOS1;Y
[1]  A THIS FUNCTION "PRE-PROCESSES" THE INVENTORY DATA BY TAKING THE
[2]  A FULL (7 X 129 X 9 X 21) INVENTORY MATRIX AND REDUCING IT TO
[3]  A THE "ADJUSTED" INVENTORY MATRIX USED BY FUNCTIONS THAT COMPUTE
[4]  A TRANSITION AND ADDITION MATRICES.
[5]  A THE ADJUSTMENTS INCLUDE:
[6]  A     ◦ REDUCING TO 5 PROGRAMS AND LOS 1 THRU 9 ONLY
[7]  A     ◦ MAKING THE LOS1 DIMENSION = (YEAR - 1). THIS FACILITATES
[8]  A     COMPUTING A LOS1 → LOS3 TRANSITION RATE.
[9]  A     ◦ PRIOR SERVICE LOS1 IS MADE THE SUM OF ALL NEW
[10] A     ACCESSION ("NEWBLOOD") P.S.
[11] A OUTPUT MATRIX IS SHAPE (5 X 69 X 9 X 9).
[12] A
[13] □ASS '17 CNA9: [GARVEYK.APL] SEPTEMBER/DA'
[14] □ASS '12 CNA9: [GARVEYK.APL] PRIDEFILE/DA'
[15] □ASS '15 CNA9: [GARVEYK.APL] PRIORSERVICE/DA'
[16] Y←88-YEAR-1
[17] A
[18] A PREVIOUS YEAR INVENTORY IS MADE LOS1 (FOR TRANSITIONS)
[19] LOS1←5 129 1+÷/ [3] □ [Y] 17
[20] →(÷/(5 129 1)÷pLOS1)/ERROR
[21] A
[22] A "M" IS THE REST OF THE INVENTORY (LOS2 THRU 21) FOR YEAR...
[23] M←2 0 0 0÷□ [88-YEAR] 17
[24] →(÷/(5 129 9 21)÷pM)/ERROR
[25] A=====
[26] A PRIDE DATA ARE USED TO GET A DISTRIBUTION OF RATING BY PROGRAM
[27] A IN LOS1. EMR DATA ARE INADEQUATE BECAUSE MOST IN LOS1 ARE GENDETS.
[28] A=====
[29] P←5 129÷÷/□ [Y] 12
[30] →(÷/(5 129)÷pP)/ERROR
[31] A
[32] PRIDE_PART←P SAFE_DIVIDE_BY ÷÷÷/P A "PRIDE_PART"=PERCENTAGE IN RATING
[33] M[1 2 3 5;;1;1]÷(÷÷÷÷/LOS1[1 2 3 5;;1] ) × PRIDE_PART[1 2 3 5;]
[34] A
[35] A "LOS1_PRIOR" ARE THE ACTUAL "NEWBLOOD" PRIOR SERVICE NUMBERS...
[36] LOS1_PRIOR←÷÷/ [2] 129 9 1÷□ [Y] 15
[37] A
[38] A PRINT "PRIOR_SERVICE" FROM "NEWBLOOD" FILE AND EMR FOR COMPARISON
[39] ÷÷÷÷÷/M[4;;1;1]
[40] ÷÷÷/LOS1_PRIOR
[41] A PLACE "NEWBLOOD" DATA AS PRIOR SERVICE, LOS 1, PAYGRADE 1 AND
[42] A ZERO OUT PAYGRADE 2 THRU 9...
[43] M[4;;1;1]÷129 1 1pLOS1_PRIOR
[44] M[:,1÷18;1]÷0
[45] A
[46] M←CONDENSE 5 129 9 9÷M A 129 RATINGS → 69 RATINGS
[47] →XIT
[48] ERROR:
[49] ' SHAPE IS INCORRECT. '
[50] →0
[51] XIT:
[52] □DAS 12 15 17
[53] MOUT←M

```

# FUNCTION: MAKE\_ALLOCATION

```

▽MOUT←MAKE_ALLOCATION YEAR
[1] MIN←80      A 1980 IS CURRENTLY THE EARLIEST DATA
[2] MAX←85      A 1985 IS CURRENTLY THE MOST RECENT YEAR (LATEST PRIDE)
[3] A
[4] →( (YEAR ≥ MIN) ^ (YEAR ≤ MAX) )/CONTINUE
[5] →YEAR_ERROR
[6] CONTINUE:
[7] A
[8] A =====
[9] A THIS FUNCTION USES THE "REDUCED" INVENTORY FILE WHERE THE LOS1
[10] A DIMENSION HAS ALREADY BEEN REPLACED BY (YEAR - 1) LOS 1 TO
[11] A PREPARE IT FOR USE IN COMPUTING TRANSITION RATES. IN ORDER TO
[12] A GET THE ACTUAL LOS1 FOR YEAR, IT MUST SELECT DATA FROM (YEAR + 1).
[13] A =====
[14] A
[15] C←ASS '12 CNA9: [GARVEYK.APL] SEPT/DA'
[16] M←[88-YEAR + 1] 12
[17] →(√/(5,MSHAPE,9,9)÷ρM)/O
[18] ALLOCATION←+/M[:,1]
[19] DAS 12
[20] A
[21] A =====
[22] A THE FOLLOWING IF "NEW-BLOOD" PRIOR SERVICE ARE TO BE INCLUDED
[23] A AS LOS1 PRIOR SERVICE IN THE ALLOCATION MATRIX...
[24] A =====
[25] A
[26] C←ASS '15 CNA9: [GARVEYK.APL] PRIORSERVICE/DA'
[27] PRIOR←[(88-YEAR)] 15
[28] →(√/(129 9 21)÷ρPRIOR)/O
[29] DAS 15
[30] P←+/+/PRIOR←CONDENSE PRIOR
[31] PRIOR_BY_LOS←ROW_PERCENT +/[2] PRIOR
[32] ALLOCATION[4;]←P
[33] A
[34] T←+/+/ALLOCATION
[35] D←(5,MSHAPE)ρT
[36] ALLOC_MATRIX←ALLOCATION ZERO_DIVIDE_BY D
[37] ' [ THE ALLOCATION MATRIX SUMS TO ',(T+/+/ALLOC_MATRIX),' ] '
[38] ' '
[39] ' '
[40] A
[41] A FORMATTING OF THE OUTPUT
[42] COLUMN←(+/ALLOCATION)÷MSHAPEρT
[43] ROW←(+/ALLOCATION)÷5ρT
[44] COLUMN←COLUMN,(+/COLUMN)

```



```

[45]  ALLOC_MATRIX←ALLOC_MATRIX, [2] 5 1ρROW
[46]  ALLOC_MATRIX←ALLOC_MATRIX, [1] (1, (MSHAPE+1))ρCOLUMN
[47]  LABEL←(4φLISTER MSHAPE), [1] 1 8ρ'TOTAL '
[48]  LABEL←LABEL, [2] ((MSHAPE+1), 4)ρ' '
[49]  HEADER←1 60ρ' RATING      4 YO      A M      5/6 YO      PRIOR      TARS      TOTA
[50]  NUMBERS←LABEL, [2] 8 0ρ*ALLOC_MATRIX * T
[51]  NUMBERS←HEADER, [1] NUMBERS
[52]  '      L O S      1      A L L O C A T I O N      F O R      19', *YEAR
[53]  ' '
[54]  NUMBERS
[55]  →XIT
[56]  MOUT←12 8ρ*ALLOC_MATRIX
[57]  SHAPE_ERROR:
[58]  ' '
[59]  ' *** ERROR *** '
[60]  ' '
[61]  ' THE DATA FILE COMPONENT READ BY THE FUNCTION IS NOT THE SHAPE '
[62]  ' THAT THE FUNCTION EXPECTS... '
[63]  ALLOC_MATRIX←1 1ρ0
[64]  →XIT
[65]  YEAR_ERROR:
[66]  ' '
[67]  ' *** ERROR **** '
[68]  ' THE YEAR SPECIFIED IS OUTSIDE THE PERIOD FOR WHICH WE HAVE DATA. '
[69]  ALLOC_MATRIX←1 1ρ0
[70]  XIT:
[71]  MOUT←12 8ρ*ALLOC_MATRIX

```

▽

**APPENDIX C**  
**SELRES DATA AND CALCULATIONS**

## APPENDIX C

### SELRES DATA AND CALCULATIONS

This appendix contains descriptions of the methodology, data, and some listings used to measure Selected Reserve (SELRES) affiliation, continuation, and inventory distributions by enlistment program. The SELRES enlistment programs include recent Navy Veterans (NAVETs) (except Active Mariners), recent Active Mariner NAVETs, Sea and Air Mariners (SAMS), and others (OTHERs). The first two reserve enlistment programs involve affiliation of prior-service active duty personnel. SAMS are non-prior-service personnel recruited directly into SELRES; their number is determined by SELRES enlistment strength planners. The OTHER category includes Advanced Paygrade (APG) personnel, other service veterans (OSVETs), and enlistments by prior service personnel not otherwise classified. The last category includes both active duty personnel separated from active duty more than 1 year before their affiliation with the reserves, and individuals whose most recent affiliation was with SELRES.

Individual active duty losses for each fiscal year are identified from the end-of-fiscal-year Enlisted Master Record (EMR) files from 1979 through 1985. The first FORTRAN program listed in annex C-1 matches these losses to CNA's longitudinal version of the Inactive Enlisted Master File (IEMF). The program then calculates affiliation rates for Active Mariners and other NAVETs separately for each rating group. Gains to SELRES from active duty are counted as affiliations of recent losses if the gain is in the same fiscal year as the loss, or in the following year. Losses with associated gains are tabulated first by enlistment program and rating group. The next section of the program calculates the inventories for recent NAVETs and Active Mariners in a form suitable for calculating average continuation rates over the FY 1979 through FY 1986 period. The final section adds non-matching individuals to the denominator for the affiliation rate computation, then calculates and prints the affiliation rates. Calculation of continuation rates proceeds by dividing total inventory in each LOS, summed across cohorts, by the total inventory in the previous LOS similarly summed. This produces the same result as computing the weighted average of continuation rates across the years. The continuation rates are presented in table C-1.

The affiliation rates for FY 1985 recent NAVETs are adjusted in ratings where the number of losses are small, to reflect a mix of rating and program-specific rates and overall affiliation behavior. Such an adjustment makes simulated affiliation behavior less susceptible to changes in the distribution of active duty losses. It is a Bayesian procedure that moderates observed rates in small cells to a greater

TABLE C-1

AVERAGE SELRES CONTINUATION  
RATES: 1979 - 1986

## RECENT NAVETS EXCEPT ACTIVE MARINER

Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8	Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8
7800 AN	0.3765	0.3976	0.5556	0.6154	0.4000	0.0000	0.0000	4400 GS	0.6406	0.7333	0.7143	0.8571	1.0000	1.0000	1.0000
6700 AB	0.4900	0.6121	0.7667	0.6071	1.0000	1.0000	1.0000	8000 HM	0.6180	0.7096	0.7845	0.8571	0.7596	0.7869	0.7818
6600 AC	0.5000	0.6506	0.7273	0.8750	0.9286	1.0000	1.0000	4300 HT	0.4547	0.6337	0.8186	0.8562	0.8250	0.9697	1.0000
6080 AF	0.4902	0.5989	0.7291	0.7548	0.8249	0.8824	0.8966	2300 IS	0.6880	0.7778	0.8636	0.8750	0.7000	0.6667	1.0000
7100 AG	0.6037	0.6505	0.7692	0.8125	0.7647	1.0000	1.0000	2600 JO	0.7059	0.7250	0.6923	0.8571	1.0000	0.8750	1.0000
7300 AK	0.5795	0.6358	0.8061	0.8103	0.9474	1.0000	1.0000	3100 LI	0.6875	0.7059	0.8889	0.8000	1.0000	1.0000	0.5000
6500 AO	0.4175	0.6739	0.6814	0.8507	0.8889	0.9333	0.6667	1750 LN	0.6190	0.7778	0.7000	1.0000	1.0000	0.5000	1.0000
7500 AS	0.4737	0.5224	0.7742	0.8500	0.6667	1.0000	0.0000	0150 MA	0.3636	0.5000	1.0000	1.0000	1.0000	0.0000	0.0000
5180 AV	0.5189	0.6550	0.7915	0.7680	0.8588	0.9072	0.9143	4700 ML	0.4545	0.6000	0.5000	0.5000	1.0000	1.0000	0.0000
6400 AW	0.4893	0.6769	0.7600	0.6739	0.8333	0.7778	0.6667	3700 MM	0.5414	0.6875	0.7857	0.8000	0.8667	0.8871	0.9545
7400 AZ	0.5498	0.5604	0.8864	0.9000	0.7368	0.9091	1.0000	0900 MN	0.6970	1.0000	0.9444	0.7857	1.0000	1.0000	1.0000
0100 BM	0.5128	0.6387	0.7895	0.7898	0.8108	0.8889	0.8571	3900 MR	0.5877	0.6460	0.7500	0.8889	0.8889	0.8182	0.6667
4000 BT	0.4577	0.5953	0.7664	0.8551	0.8250	0.8261	1.0000	2200 MS	0.5353	0.6340	0.7564	0.8687	0.8154	0.9459	0.9231
6000 CN	0.6000	0.6207	0.5556	0.3000	0.0000	0.0000	0.0000	3300 MU	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.6477	0.5106	0.8571	0.6667	0.8571	1.0000	0.6667	1400 NC	0.7500	0.6667	1.0000	1.0000	1.0000	1.0000	0.0000
1666 CTT	0.6579	0.6941	0.7708	0.8387	0.6316	0.7500	1.0000	0300 OS	0.4715	0.6167	0.7547	0.8500	0.8966	0.8636	0.8889
1633 CTM	0.7573	0.6615	0.7179	0.9091	1.0000	1.0000	1.0000	0450 OT	0.6486	0.6413	0.7170	0.8276	0.8000	0.8333	0.0000
1644 CTO	0.6726	0.6531	0.7000	0.8000	0.8182	0.8571	1.0000	2700 PC	0.4773	0.7647	1.0000	0.5000	1.0000	0.0000	0.0000
1655 CTR	0.6494	0.7312	0.7541	0.6667	0.8333	0.6154	1.0000	7600 PH	0.6405	0.6351	0.7647	0.7500	0.6667	0.0000	0.0000
1611 CTT	0.5889	0.7363	0.8033	0.8500	0.8400	0.7692	1.0000	1000 PI	0.5200	0.7059	0.8000	0.6667	1.0000	1.0000	1.0000
5080 CU	0.6162	0.7034	0.8043	0.9423	0.8974	0.7143	1.0000	4600 PM	0.4000	0.5000	1.0000	0.5000	0.0000	0.0000	0.0000
8300 DN	0.6652	0.7490	0.7963	0.8286	0.8750	0.8889	0.8889	1800 PN	0.5642	0.6163	0.7302	0.7333	0.8000	0.9167	1.0000
2100 DK	0.5625	0.8333	0.7708	0.5862	1.0000	1.0000	1.0000	7000 PR	0.4901	0.5397	0.6061	0.8333	0.8182	0.0000	0.0000
3200 DM	0.5750	0.8000	0.7692	0.6667	0.7500	1.0000	1.0000	0200 QM	0.4872	0.6400	0.7636	0.8028	0.8378	0.9333	1.0000
1900 DP	0.6091	0.6842	0.8652	0.8852	0.7667	1.0000	1.0000	1500 RM	0.5617	0.6931	0.7670	0.8253	0.8115	0.8958	0.7778
1010 DS	0.7345	0.7879	0.8649	0.8846	0.9231	0.8571	1.0000	2500 RP	0.6452	0.7500	0.7000	0.6667	0.0000	0.0000	0.0000
4100 EM	0.5496	0.6453	0.7729	0.8731	0.9135	0.8095	1.0000	3600 SN	0.3812	0.4351	0.4348	0.5455	0.7000	0.5000	0.0000
3800 EN	0.5264	0.6357	0.7676	0.7640	0.7556	0.8636	0.8750	2400 SH	0.5161	0.6378	0.7167	0.6667	0.9375	0.7000	0.6000
5380 EQ	0.6128	0.7920	0.8452	0.8448	0.9143	0.7500	0.8889	2000 SK	0.5758	0.7035	0.8203	0.8554	0.9545	0.9333	0.9375
1000 ET	0.6814	0.7302	0.7741	0.8378	0.8889	0.8769	0.8929	0250 SM	0.4127	0.6087	0.8113	0.6667	0.8462	1.0000	1.0000
0350 EW	0.6029	0.6944	0.7209	0.8846	0.8571	0.7143	0.5000	0400 ST	0.6034	0.6545	0.7745	0.8507	0.8085	0.9048	0.8750
5000 FN	0.3303	0.4545	0.6471	0.4444	0.5000	1.0000	0.0000	7200 TD	0.4328	0.3929	0.5455	0.6667	0.5000	0.0000	0.0000
0800 FT	0.5519	0.7273	0.8161	0.7686	0.8857	0.8824	0.9444	0500 TM	0.5537	0.7099	0.7826	0.7234	0.7692	0.8333	1.0000
0600 GM	0.5250	0.6885	0.7843	0.7836	0.8533	0.8500	0.9167	5800 UT	0.6054	0.7041	0.9107	0.9512	0.9500	0.8235	1.0000
								1700 YN	0.5560	0.6814	0.7550	0.7931	0.8472	0.8810	0.8333

TABLE C-1 (Continued)

## RECENT ACTIVE MARINER NAVETS

Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8	Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8
7800 AN	0.6042	0.3241	0.1729	0.2632	0.3333	0.0000	0.0000	4400 GS	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6700 AB	0.7577	0.3834	0.3907	0.6623	0.7500	0.8889	0.7143	8000 HM	0.7402	0.5537	0.4900	0.7071	0.7143	0.0000	0.7143
6600 AC	0.6081	0.4848	0.5385	0.8333	0.5000	0.0000	0.0000	4300 HT	0.7672	0.3900	0.4818	0.7474	0.8793	0.8889	0.9167
6080 AF	0.6635	0.3824	0.4280	0.6977	0.8469	0.7857	1.0000	2300 IS	0.8264	0.4634	0.5600	0.7273	1.0000	1.0000	0.0000
7100 AG	0.7279	0.4110	0.6087	0.7778	0.5000	0.5000	1.0000	2600 JO	0.6182	0.4194	0.3636	0.5000	0.5000	0.0000	0.0000
7300 AK	0.7045	0.4076	0.6250	0.7143	0.9444	1.0000	1.0000	3100 LI	0.7564	0.5577	0.3913	0.8889	1.0000	1.0000	1.0000
6500 AO	0.7184	0.3873	0.4483	0.7083	0.7586	0.7778	0.5000	1750 LN	0.8889	0.5714	0.0000	0.3000	0.0000	0.0000	0.0000
7500 AS	0.6964	0.4098	0.5500	0.7895	0.8333	0.8000	1.0000	0150 MA	1.0000	1.0000	0.5000	0.0000	0.0000	0.0000	0.0000
6180 AV	0.7368	0.3640	0.4488	0.6545	0.8070	0.8519	0.7778	4700 ML	0.6667	0.4375	0.1429	1.0000	0.0000	0.0000	0.0000
6400 AW	0.7184	0.4679	0.5172	0.8261	0.9333	0.8571	0.5000	3700 MM	0.7463	0.3574	0.3289	0.7018	0.8148	0.0000	1.0000
7400 AZ	0.7545	0.3488	0.3714	0.9091	0.7143	0.8000	0.5000	0900 MN	0.6964	0.4706	0.6429	0.8571	0.7500	1.0000	0.0000
0100 BM	0.7997	0.4545	0.4974	0.7483	0.7356	0.8667	0.9333	3900 MR	0.7435	0.4087	0.4110	0.7600	0.8750	1.0000	1.0000
4000 BT	0.7612	0.3471	0.3782	0.7037	0.8000	0.6667	0.5000	2200 MS	0.7681	0.4472	0.5083	0.7750	0.8409	0.7600	0.9091
6000 CN	1.0000	0.4545	0.3000	0.5556	0.2000	0.0000	0.0000	3300 MU	0.6667	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CT	0.7027	0.5000	0.6364	0.8571	1.0000	1.0000	1.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CT	0.8333	0.8889	0.5714	1.0000	0.3333	0.0000	0.0000	0300 OS	0.7430	0.4712	0.4151	0.6897	0.8125	0.8571	0.6667
1633 CTM	0.6667	0.3750	0.5000	1.0000	1.0000	0.0000	0.0000	0450 OT	0.6970	0.3000	0.3333	1.0000	0.0000	0.0000	0.0000
1644 CTO	0.6667	0.3929	0.6500	0.5455	1.0000	1.0000	1.0000	2700 PC	0.8235	0.4737	0.4359	0.6875	0.7500	0.5000	1.0000
1655 CTR	0.7800	0.5484	0.5294	0.7778	0.5000	0.3333	1.0000	7600 PH	0.7120	0.4628	0.4706	0.4545	0.7143	1.0000	0.0000
1611 CTT	0.7018	0.5000	0.2941	0.7500	1.0000	1.0000	0.0000	1080 PI	0.7377	0.2821	0.4444	1.0000	0.6667	0.0000	0.0000
5080 CU	0.7714	0.4844	0.5684	0.8478	0.7826	0.8182	0.0000	4600 PM	0.7667	0.3043	0.6667	0.3333	1.0000	1.0000	0.0000
8300 DN	0.8026	0.5556	0.5455	0.7273	0.8462	1.0000	0.6667	1800 PN	0.7133	0.5036	0.5440	0.7213	0.6875	0.5833	0.7500
2100 DK	0.8212	0.5159	0.4333	0.7200	0.6923	0.7143	0.0000	7000 PR	0.6724	0.4028	0.5556	0.5000	0.8000	1.0000	1.0000
3200 DM	1.0000	0.7037	0.4118	0.7143	0.8000	1.0000	1.0000	0200 QM	0.7469	0.4230	0.5049	0.7209	0.6364	1.0000	0.8000
1900 DP	0.8158	0.3883	0.5429	0.6667	0.7500	0.8333	1.0000	1500 RM	0.7361	0.4118	0.5614	0.7876	0.8590	0.9400	0.9412
1010 DS	0.7317	0.3333	0.3750	1.0000	0.6667	1.0000	0.6667	2500 RP	0.6111	0.6250	0.5000	1.0000	1.0000	1.0000	0.0000
4100 EM	0.7840	0.3789	0.4498	0.7701	0.8393	0.8214	0.9231	3600 SN	0.6138	0.3507	0.2451	0.3529	0.6250	0.1667	0.0000
3800 EN	0.7363	0.3881	0.4050	0.6471	0.6667	0.6923	1.0000	2400 SH	0.7251	0.4449	0.4330	0.7027	0.6875	1.0000	1.0000
5380 EO	0.7472	0.4363	0.5426	0.7949	1.0000	0.8636	1.0000	2000 SK	0.7722	0.5149	0.5842	0.7600	0.8727	0.7000	1.0000
1000 ET	0.9159	0.2996	0.5075	0.7083	0.8462	0.5000	1.0000	0250 SM	0.7383	0.3978	0.3770	0.5500	0.8750	1.0000	1.0000
0350 EW	0.8105	0.3333	0.5000	0.8000	1.0000	1.0000	0.0000	0400 ST	0.7571	0.4013	0.5192	0.6957	0.7000	0.7500	1.0000
5000 FN	0.6116	0.3452	0.2754	0.5556	0.3636	0.7500	0.0000	7200 TD	0.6812	0.3953	0.2857	0.2500	1.0000	1.0000	0.0000
0800 FT	0.8038	0.4278	0.4561	0.6316	0.6667	1.0000	0.0000	0500 TM	0.7604	0.4338	0.5625	0.8696	0.8333	1.0000	0.8000
0600 GM	0.7748	0.3940	0.4867	0.7857	0.7429	0.8750	0.8750	5800 UT	0.7650	0.4667	0.4884	0.7500	0.6000	0.0000	0.0000
								1700 YN	0.7061	0.4792	0.5048	0.7093	0.6226	0.7619	0.8333

TABLE C-1 (Continued)

## SAM PROGRAM

Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8	Rate	1-2	2-3	3-4	4-5	5-6	6-7	7-8
7800 AN	0.7800	0.6300	0.3800	0.3200	0.3100	0.1300	0.0000	4400 GS	0.0000	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000
6700 AB	0.6500	0.8500	0.0000	0.0000	1.0000	1.0000	0.0000	8000 HM	0.8000	0.9000	0.9400	0.9600	0.8000	0.4700	0.8300
6600 AC	0.8100	0.8000	0.0000	0.0000	0.0000	0.0000	0.0000	4300 HT	0.7800	1.0000	0.9000	1.0000	0.7500	0.0000	1.0000
6080 AF	0.7400	0.7300	0.8300	0.7300	0.6800	0.2600	0.5000	2300 IS	0.7700	0.5000	1.0000	0.0000	0.0000	0.0000	0.0000
7100 AG	0.8500	0.9000	0.8500	0.7500	1.0000	1.0000	1.0000	2600 JO	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7300 AK	0.8400	1.0000	0.5000	0.0000	1.0000	0.4000	0.0000	3100 LI	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6500 AO	0.7300	0.8500	0.8200	0.6400	0.6600	0.7500	0.5000	1750 LN	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
7500 AS	1.0000	1.0000	0.0000	0.0000	1.0000	0.0000	1.0000	0150 MA	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6180 AV	0.8100	0.9400	0.8300	0.8200	0.8300	0.4000	0.5000	4700 ML	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
6400 AW	0.7400	1.0000	0.8300	0.9000	0.8700	0.5700	1.0000	3700 MM	0.6000	1.0000	0.6600	0.8300	0.5000	0.0000	0.0000
7400 AZ	0.8200	1.0000	0.5000	0.0000	1.0000	0.5000	0.0000	0900 MN	0.7700	1.0000	1.0000	1.0000	1.0000	0.0000	0.0000
0100 BM	0.6600	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000	3900 MR	0.7700	0.6600	0.0000	0.0000	0.0000	0.0000	0.0000
4000 BT	0.2500	1.0000	0.5500	0.6600	0.8500	0.0000	0.0000	2200 MS	0.7300	0.6600	0.8000	0.7800	0.9200	0.4500	0.6600
6000 CN	0.7800	0.1800	0.0200	0.0300	0.0000	0.0000	0.0000	3300 MJ	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1622 CTA	0.6600	0.8500	0.0000	0.0000	0.0000	0.0000	0.0000	1400 NC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1666 CTT	0.7500	0.0000	0.0000	0.0000	0.0000	1.0000	0.0000	0300 OS	0.7800	0.8700	0.7000	0.8300	0.6200	0.2800	1.0000
1633 CTM	0.8100	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0450 OT	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1644 CTO	0.7800	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	2700 PC	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1655 CTR	0.8100	0.8500	0.0000	0.0000	0.0000	0.5000	0.0000	7600 PH	0.6600	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
1611 CTT	0.7500	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	1080 PI	1.0000	0.8300	0.0000	0.0000	0.0000	0.0000	0.0000
5080 CU	0.7800	0.8900	0.8100	0.8200	0.8900	0.4200	0.6000	4600 PM	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
8300 DN	0.8000	1.0000	0.0000	0.0000	1.0000	0.0000	0.0000	1800 PN	0.7500	0.8700	0.6600	1.0000	1.0000	0.0000	0.0000
2100 DK	0.8800	0.6600	0.0000	1.0000	0.0000	0.0000	0.0000	7000 PR	1.0000	0.6600	0.5000	0.8800	0.5000	0.0000	0.0000
3200 DM	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0200 QM	0.7500	0.9400	0.8500	1.0000	1.0000	0.0000	0.0000
1900 DP	0.8400	1.0000	0.0000	1.0000	0.0000	0.0000	0.0000	1500 RM	0.7700	0.9000	0.7600	0.7000	0.9000	0.3300	0.0000
1010 DS	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000	2500 RP	0.8100	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
4100 EM	0.7600	1.0000	1.0000	0.0000	0.0000	0.0000	0.0000	3600 SN	0.7800	0.6500	0.4700	0.3200	0.2200	0.2500	0.0000
3800 EN	0.7600	0.7700	0.7800	0.8300	0.7300	0.2500	0.0000	2490 SH	0.5800	0.7500	1.0000	1.0000	0.0000	0.0000	0.0000
5380 EQ	0.8100	0.8900	0.8000	0.7300	0.8200	0.4000	0.7700	2000 SK	0.8100	1.0000	0.6300	0.8500	0.8800	0.6600	0.0000
1000 ET	0.8300	1.0000	0.6600	0.0000	1.0000	0.6600	0.3300	0250 SM	0.8000	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
0350 EW	0.7200	0.8200	0.8700	0.8300	1.0000	0.0000	0.0000	0400 ST	0.8000	0.8700	0.6700	0.8700	0.8500	0.6600	0.0000
5000 FN	0.7800	0.7100	0.5000	0.5000	0.3000	0.0000	0.0000	7200 TD	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0800 FT	0.7700	1.0000	0.7000	0.6100	0.6600	0.3300	0.0000	0500 TM	0.8600	1.0000	0.0000	0.0000	0.0000	0.0000	0.0000
0600 GM	0.8200	1.0000	0.9400	0.8600	0.7700	0.7100	0.0000	5800 UT	0.8000	0.9800	0.9200	0.9000	0.8000	0.4700	1.0000
								1700 YN	0.8400	0.8800	0.7000	0.5000	0.8700	1.0000	0.0000

extent than large cells.<sup>1</sup> Within each program, GENDETs are adjusted separately from the other ratings because their affiliation rates are much lower. The adjustment is applied to the calculated affiliation rates in several steps for the NAVET and Active Mariner rates separately.

First, a two-parameter beta probability distribution is fit to observed affiliation behavior and these parameters are then applied to the observed affiliation rates as in equation C-1:

$$a = \frac{\alpha + X}{\alpha + \beta + N} \quad (C-1)$$

where  $a$  is the adjusted rate,  $X/N$  is the observed rate, and  $\alpha/(\alpha + \beta)$  is the mean of the fitted beta distribution. The adjustment is always toward the overall mean, and has its biggest effect when the cell size ( $N$ =losses) is small, and when  $X/N$  differs from the fitted mean of the distribution. Table C-2 shows adjusted and unadjusted affiliation rates based on the behavior of FY 1985 active duty losses.

Annex C-2 presents the program listings used to determine the structure of the FY 1985 SELRES inventory by enlistment program, rating group, and LOS. Because the IEMF carries no date field for beginning of SELRES affiliation, the LOS must be computed from the difference between the observation date and the first time the individual appears as a gain to SELRES. SAM LOS 1 inventories by rating are determined by looking ahead one year to find the ratings to which SAM GENDET accessions are distributed. This is consistent with the calculation of continuation rates for SAMs. The resulting inventory appears in table C-3.

Tabulations of the September 1985 SELRES inventory paygrade structure by rating group is presented in table C-4. This is done for the purpose of converting the inventory by LOS dimension to an inventory dimensioned by paygrade for comparison with the Reserve Enlisted Programmed Authorization (EPA). This distribution will be sensitive to changes in accessions among enlistment programs, especially SAMs. Table C-5 contains the allocation of SAM and OTHER accessions to ratings. The SAM allocations can be changed during the simulation. The reserve EPA is listed separately for each year, by rating group, in table C-6.

---

1. For additional discussion of the technique, see John D. Hey, Data in Doubt, Basil Blackwell Ltd: Oxford, 1983.

TABLE C-2

FY 1985 SELRES AFFILIATION RATES BY  
RATING AND PROGRAM

## NAVET

## ACTIVE MARINER

Rating	Aff	Loss	Actual	Adjusted	Aff	Loss	Actual	Adjusted
AN	36	2817	0.0128	0.0133	257	1058	0.2429	0.2414
AB	67	1032	0.0649	0.0660	141	184	0.7663	0.7578
AC	32	254	0.1260	0.1204	16	23	0.6957	0.6841
AF	203	3037	0.0668	0.0672	323	563	0.5737	0.5766
AG	49	242	0.2025	0.1841	30	33	0.9091	0.8257
AK	58	499	0.1162	0.1140	32	64	0.5000	0.5364
AO	36	749	0.0481	0.0506	149	209	0.7129	0.7095
AS	26	298	0.0872	0.0877	42	57	0.7368	0.7209
AV	255	2907	0.0877	0.0878	301	446	0.6749	0.6747
AW	59	359	0.1643	0.1557	31	48	0.6458	0.6520
AZ	49	426	0.1150	0.1126	46	77	0.5974	0.6107
BM	186	1602	0.1161	0.1154	349	403	0.8660	0.8578
BT	78	1791	0.0436	0.0448	144	237	0.6076	0.6118
CN	0	75	0.0000	0.0136	0	27	0.0000	0.1520
CTA	12	81	0.1481	0.1268	2	6	0.3333	0.5836
CTI	20	128	0.1563	0.1384	1	4	0.2500	0.5913
CTM	25	199	0.1256	0.1188	3	5	0.6000	0.6537
CTO	30	170	0.1765	0.1577	3	6	0.5000	0.6260
CTR	24	164	0.1463	0.1338	9	13	0.6923	0.6789
CTT	24	202	0.1188	0.1134	4	9	0.4444	0.5930
CU	43	549	0.0783	0.0793	55	82	0.6707	0.6704
DT	70	320	0.2188	0.2022	26	44	0.5909	0.6132
DK	26	273	0.0952	0.0945	20	28	0.7143	0.6968
DM	5	44	0.1136	0.1016	6	7	0.8571	0.7226
DP	87	575	0.1513	0.1467	24	35	0.6857	0.6801
DS	29	422	0.0687	0.0709	2	10	0.2000	0.4990
EM	186	2743	0.0678	0.0682	150	198	0.7576	0.7504
EN	114	1283	0.0889	0.0889	193	256	0.7539	0.7485
EQ	34	472	0.0720	0.0737	91	127	0.7165	0.7108
ET	146	2021	0.0722	0.0727	42	74	0.5676	0.5870
EW	27	310	0.0871	0.0875	29	42	0.6905	0.6841
FN	57	2725	0.0209	0.0186	314	1067	0.2943	0.2899
FT	70	1128	0.0621	0.0632	90	120	0.7500	0.7397
GM	78	838	0.0931	0.0929	174	213	0.8169	0.8056
GS	21	321	0.0654	0.0686	2	8	0.2500	0.5380
HM	552	2872	0.1922	0.1905	170	262	0.6489	0.6501
HT	189	1806	0.1047	0.1043	250	331	0.7553	0.7509
IS	17	110	0.1545	0.1352	17	24	0.7083	0.6917
JO	10	78	0.1282	0.1139	4	9	0.4444	0.5930
LI	5	54	0.0926	0.0916	6	9	0.6667	0.6682
LN	9	45	0.2000	0.1437	1	1	1.0000	0.6868
MA	1	118	0.0085	0.0320	1	2	0.5000	0.6518
ML	2	31	0.0645	0.0802	3	6	0.5000	0.6260
MM	179	4141	0.0432	0.0438	128	215	0.5953	0.6009
MN	7	56	0.1250	0.1091	10	15	0.6667	0.6679
MR	38	368	0.1033	0.1018	61	79	0.7722	0.7534



TABLE C-2 (Continued)

Rating	NAVET				ACTIVE MARINER			
	Aff	Loss	Actual	Adjusted	Aff	Loss	Actual	Adjusted
MS	195	2542	0.0767	0.0770	141	260	0.5423	0.5503
MU	0	79	0.0000	0.0340	0	0	*****	0.6690
NC	0	113	0.0000	0.0268	0	0	*****	0.6690
OS	168	1748	0.0961	0.0960	148	221	0.6697	0.6696
OT	15	214	0.0701	0.0738	5	9	0.5556	0.6306
PC	5	136	0.0368	0.0507	20	21	0.9524	0.8233
PH	50	341	0.1466	0.1398	9	13	0.6923	0.6789
PI	9	116	0.0776	0.0813	7	11	0.6364	0.6565
PM	0	6	0.0000	0.0803	0	0	*****	0.6690
PN	67	754	0.0889	0.0890	41	89	0.4607	0.4950
PR	14	175	0.0800	0.0822	10	24	0.4167	0.5234
QM	55	582	0.0945	0.0942	82	116	0.7069	0.7019
RM	243	1956	0.1242	0.1234	110	174	0.6322	0.6356
RP	28	126	0.2222	0.1861	6	11	0.5455	0.6215
SN	135	10322	0.0131	0.0132	479	2705	0.1771	0.1780
SH	61	728	0.0838	0.0842	36	62	0.5806	0.6002
SK	111	948	0.1171	0.1158	141	213	0.6620	0.6625
SM	35	437	0.0801	0.0811	42	75	0.5600	0.5807
ST	30	861	0.0348	0.0378	39	70	0.5571	0.5796
TD	4	330	0.0121	0.0220	6	19	0.3158	0.4855
TM	64	564	0.1135	0.1117	40	61	0.6557	0.6587
UT	22	323	0.0681	0.0710	48	66	0.7273	0.7150
YN	136	1443	0.0942	0.0941	56	131	0.4275	0.4561

BETA DISTRIBUTION PARAMETERS  
(ALPHA, BETA)

	NAVET	AM
GENDET	(20.52, 1,413.79)	(13.832, 50.15)
RATED	(4.30, 43.21)	(11.76, 5.82)

TABLE C-3

1985 SELRES INVENTORY BY  
ENLISTMENT PROGRAM, RATING AND LOS

RECENT NAVETS EXCEPT ACTIVE MARINER

Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7	Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7
7800 AN	26.	14.	5.	5.	2.	1.	0.	4400 GS	13.	10.	6.	5.	0.	0.	1.
6700 AB	50.	26.	24.	7.	4.	4.	4.	8000 HM	498.	314.	237.	195.	173.	106.	49.
6600 AC	28.	13.	13.	8.	7.	3.	2.	4300 HT	177.	78.	42.	62.	38.	23.	11.
6000 AF	190.	105.	71.	84.	80.	56.	26.	2300 IS	14.	20.	7.	8.	9.	3.	2.
7100 AG	48.	24.	9.	11.	9.	4.	1.	2600 JO	10.	7.	6.	4.	1.	3.	2.
7300 AK	53.	24.	26.	32.	8.	5.	6.	3100 LI	5.	4.	2.	1.	1.	0.	2.
6500 AO	48.	18.	17.	23.	20.	11.	2.	1750 LN	11.	4.	3.	0.	3.	1.	0.
7500 AS	23.	9.	9.	9.	2.	4.	1.	0150 MA	0.	1.	0.	1.	1.	0.	0.
6180 AV	250.	94.	61.	87.	69.	65.	30.	4700 ML	4.	0.	2.	1.	0.	1.	0.
6400 AW	54.	23.	20.	10.	15.	5.	3.	3700 MM	138.	89.	68.	67.	44.	36.	19.
7400 AZ	45.	13.	14.	11.	7.	5.	7.	0900 MN	8.	0.	3.	4.	4.	2.	2.
0100 BM	178.	84.	45.	59.	47.	12.	7.	3900 MR	37.	17.	18.	15.	5.	8.	3.
4000 BT	104.	36.	15.	21.	12.	11.	9.	2200 MS	150.	65.	30.	24.	22.	23.	9.
6000 CN	1.	0.	3.	3.	2.	0.	0.	3300 MU	0.	0.	0.	0.	0.	0.	0.
1022 CTA	15.	5.	4.	3.	2.	1.	2.	1400 NC	0.	0.	0.	0.	0.	1.	0.
1666 CTT	20.	17.	6.	8.	5.	5.	2.	0300 OS	146.	32.	28.	32.	31.	13.	10.
1633 CTM	14.	8.	7.	11.	3.	3.	2.	0450 OT	8.	10.	12.	4.	4.	11.	0.
1644 CTO	19.	19.	15.	11.	3.	4.	2.	2700 PC	6.	3.	5.	2.	1.	1.	0.
1655 CTR	25.	8.	13.	12.	4.	9.	3.	7600 PH	43.	23.	18.	7.	2.	0.	1.
1611 CTT	27.	8.	12.	11.	10.	5.	6.	1080 PI	9.	8.	4.	4.	2.	1.	4.
5080 CU	42.	21.	26.	11.	10.	15.	8.	4600 PM	2.	3.	1.	1.	0.	0.	0.
8300 DN	73.	38.	35.	50.	10.	14.	8.	1800 PN	41.	62.	22.	34.	21.	13.	4.
2100 DK	11.	9.	12.	16.	3.	0.	4.	7000 PR	19.	6.	7.	5.	9.	0.	0.
3200 DM	8.	6.	0.	4.	0.	2.	0.	0200 QM	51.	33.	19.	26.	16.	6.	8.
1900 DP	85.	40.	18.	25.	9.	10.	3.	1500 RM	237.	125.	89.	104.	68.	67.	24.
1010 DS	24.	15.	8.	12.	6.	3.	4.	2500 RP	27.	10.	4.	5.	0.	0.	0.
4100 EM	162.	100.	68.	77.	36.	36.	19.	3600 SN	113.	56.	25.	14.	4.	2.	1.
3800 EN	117.	62.	31.	33.	16.	12.	7.	2490 SH	59.	33.	25.	4.	5.	2.	5.
5380 EQ	38.	21.	20.	17.	12.	9.	7.	2000 SK	102.	46.	54.	88.	35.	13.	14.
1000 ET	131.	75.	60.	43.	43.	31.	26.	0250 SM	33.	7.	11.	16.	1.	5.	4.
0350 EW	21.	9.	7.	10.	4.	5.	1.	0400 ST	33.	40.	16.	12.	20.	10.	7.
5000 FN	48.	12.	2.	6.	1.	1.	0.	7200 TD	6.	0.	0.	1.	0.	1.	0.
0800 FT	69.	33.	31.	32.	27.	14.	16.	0500 TM	64.	33.	9.	14.	0.	4.	2.
0600 GM	95.	30.	36.	44.	30.	25.	12.	5800 UT	28.	17.	12.	21.	3.	12.	3.
								1700 YN	102.	121.	68.	57.	25.	23.	19.

TABLE C-3 (Continued)

## RECENT ACTIVE MARINER NAVETS

Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7	Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7
7800 AN	214.	131.	26.	3.	0.	1.	0.	4400 GS	2.	0.	0.	0.	0.	0.	0.
6700 AB	118.	93.	24.	9.	20.	10.	6.	8000 HM	154.	113.	99.	34.	17.	13.	7.
6600 AC	14.	4.	3.	0.	0.	2.	0.	4300 HT	225.	182.	58.	20.	27.	13.	10.
6080 AF	300.	282.	80.	31.	41.	30.	4.	2300 IS	18.	24.	5.	1.	4.	3.	0.
7100 AG	27.	17.	9.	4.	1.	1.	1.	2600 JO	5.	1.	1.	2.	1.	0.	0.
7300 AK	29.	36.	13.	8.	6.	6.	4.	3100 LI	9.	10.	4.	1.	5.	0.	1.
6500 AO	132.	76.	15.	7.	13.	7.	2.	1750 LN	1.	3.	1.	0.	0.	0.	0.
7500 AS	41.	25.	4.	3.	5.	3.	1.	0150 MA	1.	0.	1.	0.	0.	0.	0.
6180 AV	290.	185.	43.	22.	19.	15.	9.	4700 ML	3.	2.	2.	1.	0.	0.	0.
6400 AW	29.	25.	9.	4.	7.	5.	2.	3700 NM	114.	113.	49.	17.	10.	7.	3.
7400 AZ	40.	19.	10.	4.	0.	3.	2.	0900 NN	8.	2.	2.	0.	1.	1.	1.
0100 BM	306.	198.	88.	34.	25.	27.	13.	3900 MR	59.	41.	10.	4.	6.	2.	5.
4000 BT	125.	131.	44.	13.	2.	3.	2.	2200 MS	114.	69.	42.	24.	16.	10.	11.
6000 CN	1.	2.	1.	0.	1.	0.	0.	3300 MU	0.	0.	0.	0.	0.	0.	0.
1622 CTA	3.	2.	0.	0.	2.	3.	1.	1400 NC	0.	0.	0.	0.	0.	0.	0.
1666 CTI	0.	1.	1.	0.	1.	0.	0.	0300 OS	130.	86.	38.	6.	6.	4.	3.
1633 CTM	0.	1.	1.	0.	1.	0.	0.	0450 OT	4.	4.	3.	2.	0.	0.	0.
1644 CTO	3.	7.	5.	3.	2.	1.	0.	2700 PC	17.	18.	6.	6.	2.	2.	2.
1655 CTR	9.	1.	2.	1.	1.	1.	1.	7600 PH	12.	7.	3.	3.	2.	3.	0.
1611 CTT	3.	8.	5.	1.	1.	2.	0.	1080 PI	7.	5.	3.	0.	3.	0.	0.
5080 CU	52.	58.	21.	18.	7.	10.	1.	4600 PM	1.	2.	1.	0.	0.	0.	0.
8300 DN	25.	16.	7.	5.	5.	4.	3.	1800 PN	30.	33.	13.	11.	13.	5.	2.
2100 DK	19.	9.	10.	8.	3.	5.	1.	7000 PR	9.	10.	5.	5.	2.	1.	2.
3200 DM	5.	3.	0.	1.	2.	1.	1.	0200 QM	71.	56.	19.	13.	7.	3.	2.
1900 DP	22.	13.	4.	3.	1.	2.	3.	1500 RM	94.	50.	21.	13.	21.	32.	5.
1010 DS	5.	7.	1.	1.	0.	0.	3.	2500 RP	7.	2.	0.	0.	1.	14.	14.
4100 EM	139.	130.	37.	14.	21.	11.	9.	3600 SN	411.	188.	65.	18.	5.	1.	0.
3800 EN	168.	122.	37.	13.	15.	6.	3.	2490 SH	30.	37.	17.	12.	6.	3.	0.
5380 EQ	82.	36.	19.	6.	6.	14.	6.	2600 SK	108.	89.	30.	25.	18.	19.	3.
1000 ET	43.	30.	17.	5.	5.	4.	1.	0250 SM	37.	22.	15.	9.	2.	1.	5.
0350 EW	24.	10.	2.	2.	0.	2.	0.	0400 ST	37.	36.	15.	8.	3.	2.	0.
5000 FN	260.	134.	41.	10.	2.	3.	0.	7200 TD	7.	8.	2.	0.	0.	1.	0.
0800 FT	77.	47.	14.	5.	1.	3.	0.	0500 TM	33.	35.	13.	9.	0.	1.	5.
0600 GM	177.	117.	50.	12.	14.	4.	7.	5800 UT	44.	37.	11.	8.	4.	1.	0.
								1700 YN	40.	67.	41.	14.	16.	12.	4.

TABLE C-3 (Continued)

## SAM PROGRAM

Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7	Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7
7800 AN	774.	444.	55.	52.	57.	29.	3.	4400 GS	0.	11.	1.	0.	0.	0.	0.
6700 AB	41.	20.	0.	0.	2.	1.	0.	8000 HM	920.	490.	52.	27.	21.	0.	0.
6600 AC	38.	5.	0.	0.	0.	0.	0.	4300 HT	38.	23.	10.	2.	4.	23.	6.
6080 AF	101.	138.	106.	109.	100.	67.	6.	2300 IS	18.	2.	2.	1.	0.	0.	1.
7100 AG	7.	11.	7.	4.	2.	3.	1.	2600 JO	0.	0.	0.	0.	0.	0.	0.
7300 AK	59.	7.	4.	0.	2.	5.	0.	3100 LI	0.	0.	0.	0.	0.	0.	0.
6500 AO	34.	47.	35.	17.	9.	4.	2.	1750 LN	0.	0.	0.	0.	0.	0.	0.
7500 AS	3.	5.	0.	0.	1.	1.	1.	0150 MA	0.	0.	0.	0.	0.	0.	0.
6180 AV	180.	196.	96.	95.	68.	30.	2.	4700 ML	3.	0.	0.	0.	0.	0.	0.
6400 AW	35.	43.	30.	20.	16.	7.	1.	3700 MN	5.	8.	18.	6.	4.	0.	0.
7400 AZ	35.	4.	2.	0.	2.	2.	0.	0900 MR	35.	20.	4.	2.	1.	2.	0.
0100 BA	3.	1.	1.	0.	0.	0.	0.	3900 MS	18.	3.	0.	0.	1.	0.	0.
4000 BT	4.	3.	9.	3.	7.	1.	0.	2200 MU	142.	68.	35.	14.	25.	11.	3.
6000 CN	14.	33.	40.	28.	29.	11.	0.	3300 NC	0.	0.	0.	0.	0.	0.	0.
1622 CTA	9.	21.	0.	0.	0.	0.	0.	1400 OS	0.	0.	0.	0.	0.	0.	0.
1666 CTH	8.	0.	0.	0.	0.	1.	0.	0300 OT	148.	79.	31.	6.	8.	7.	1.
1633 CTM	11.	0.	0.	0.	0.	0.	0.	0450 PC	0.	0.	0.	0.	0.	0.	0.
1644 CTO	46.	67.	0.	0.	1.	1.	1.	2700 PH	0.	0.	0.	0.	0.	0.	0.
1655 CTR	37.	40.	0.	0.	1.	2.	0.	7600 PI	6.	8.	0.	0.	0.	0.	0.
1611 CTT	4.	0.	0.	0.	0.	0.	0.	1080 PM	5.	6.	0.	0.	0.	0.	0.
5080 CU	337.	244.	189.	116.	91.	71.	15.	4600 PN	0.	0.	0.	0.	0.	0.	0.
8300 DN	21.	4.	0.	0.	1.	0.	0.	1800 PR	16.	40.	3.	2.	0.	0.	0.
2100 DK	9.	3.	0.	0.	0.	0.	0.	7000 QM	3.	3.	4.	9.	2.	2.	0.
3200 DM	1.	0.	0.	0.	0.	0.	0.	0200 RM	12.	17.	7.	5.	4.	0.	0.
1900 DP	13.	17.	0.	0.	0.	0.	0.	1500 RP	217.	106.	21.	17.	10.	6.	0.
1010 DS	2.	1.	0.	0.	0.	1.	0.	2500 SN	11.	0.	0.	0.	0.	0.	0.
4100 EM	25.	23.	1.	0.	0.	0.	0.	3600 SH	3292.	2045.	90.	40.	57.	27.	3.
3800 EN	26.	44.	14.	6.	19.	8.	1.	2400 SK	12.	4.	1.	1.	0.	0.	0.
5380 EQ	392.	330.	233.	175.	146.	99.	9.	2000 SM	205.	119.	19.	7.	9.	3.	1.
1000 ET	48.	18.	3.	1.	4.	3.	3.	0250 ST	36.	9.	2.	1.	0.	1.	0.
0350 EW	18.	29.	8.	6.	4.	4.	0.	0400 TD	46.	24.	31.	8.	7.	3.	0.
5000 FN	816.	666.	6.	6.	10.	3.	0.	7200 TM	0.	0.	0.	0.	0.	0.	0.
0800 FT	61.	26.	34.	13.	6.	3.	1.	0500 UT	15.	11.	0.	0.	0.	1.	0.
0600 GW	68.	33.	18.	15.	9.	7.	0.	5800 YN	184.	137.	114.	84.	90.	42.	7.
									26.	53.	10.	4.	8.	1.	0.

TABLE C-3 (Continued)

Rate	OTHER RESERVE														
	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7	Rate	LOS 1	LOS 2	LOS 3	LOS 4	LOS 5	LOS 6	LOS 7
7800 AN	208.	123.	74.	38.	14.	8.	14.	4400 GS	11.	7.	5.	5.	6.	0.	1.
6700 AB	85.	43.	37.	10.	8.	6.	170.	8000 HM	589.	436.	493.	286.	202.	137.	1154.
6600 AC	27.	22.	22.	22.	6.	4.	38.	4300 HT	270.	192.	241.	163.	114.	118.	778.
6000 AF	334.	224.	183.	141.	134.	150.	1218.	2300 IS	124.	84.	58.	53.	37.	65.	212.
7100 AG	30.	18.	26.	17.	17.	10.	64.	2600 JO	26.	9.	32.	12.	7.	5.	51.
7300 AK	108.	114.	92.	51.	21.	18.	302.	3100 LI	5.	3.	9.	1.	6.	6.	47.
6500 AO	57.	46.	48.	51.	28.	21.	193.	1750 LN	23.	18.	9.	6.	3.	5.	32.
7500 AS	32.	20.	17.	13.	10.	27.	75.	0150 MA	78.	92.	4.	16.	2.	40.	25.
6180 AV	284.	201.	168.	150.	99.	117.	1044.	4700 ML	4.	4.	5.	4.	6.	4.	15.
6400 AW	35.	30.	18.	30.	14.	13.	253.	3700 MM	92.	111.	134.	148.	55.	57.	606.
7400 AZ	66.	26.	21.	24.	13.	7.	160.	0900 MN	10.	8.	14.	12.	14.	6.	131.
0100 BM	273.	189.	249.	189.	111.	91.	899.	3900 MR	105.	89.	57.	29.	15.	44.	299.
4000 BT	131.	53.	52.	57.	25.	53.	372.	2200 MS	247.	200.	144.	141.	64.	84.	547.
6000 CN	23.	49.	43.	15.	5.	1.	10.	3300 MU	0.	0.	1.	1.	0.	0.	0.
1622 CTA	31.	20.	22.	13.	8.	14.	81.	1400 NC	0.	0.	0.	0.	0.	0.	2.
1666 CTI	16.	13.	14.	10.	9.	13.	45.	0300 OS	118.	73.	96.	69.	50.	38.	348.
1633 CTM	21.	13.	15.	10.	11.	13.	127.	0450 OT	2.	2.	13.	12.	3.	3.	14.
1644 CTO	23.	20.	24.	21.	15.	5.	135.	2700 PC	9.	8.	7.	3.	3.	6.	84.
1655 CTR	30.	24.	32.	20.	16.	14.	187.	7600 PH	35.	20.	23.	22.	13.	8.	145.
1611 CTT	24.	13.	23.	15.	8.	8.	105.	1000 PI	25.	16.	20.	12.	11.	11.	65.
5000 CU	237.	273.	312.	216.	221.	250.	1423.	4600 PM	2.	4.	4.	4.	0.	3.	9.
8300 DN	75.	82.	33.	52.	27.	16.	203.	1800 PN	46.	66.	55.	54.	36.	22.	372.
2100 DK	15.	13.	21.	14.	5.	7.	147.	7000 PR	24.	10.	7.	8.	5.	8.	32.
3200 DM	21.	9.	10.	6.	7.	13.	74.	0200 QM	73.	69.	79.	54.	40.	26.	279.
1900 DP	135.	78.	44.	67.	42.	33.	227.	1500 RM	270.	184.	229.	174.	117.	106.	748.
1010 DS	27.	15.	19.	14.	10.	22.	52.	2500 RP	2.	9.	3.	1.	2.	1.	3.
4100 EM	200.	165.	133.	161.	68.	52.	701.	3600 SN	1173.	541.	270.	84.	30.	16.	23.
3800 EN	142.	100.	76.	71.	56.	55.	605.	2490 SH	78.	76.	53.	16.	11.	12.	186.
5300 EQ	257.	320.	363.	262.	261.	312.	1720.	2000 SK	568.	457.	456.	284.	95.	123.	1237.
1000 ET	167.	154.	187.	125.	89.	81.	590.	0250 SM	44.	40.	39.	27.	21.	14.	141.
0350 EW	18.	12.	8.	7.	5.	2.	28.	0400 ST	48.	36.	28.	26.	18.	12.	142.
5000 FN	198.	116.	61.	30.	20.	9.	19.	7200 TD	4.	3.	1.	2.	0.	1.	13.
0800 FT	65.	59.	61.	40.	39.	28.	247.	0500 TM	38.	34.	34.	33.	17.	7.	141.
0600 GM	115.	107.	135.	100.	50.	36.	340.	5800 UT	217.	188.	137.	116.	105.	138.	797.
								1700 YN	204.	512.	492.	214.	106.	102.	1451.

Table C-4

PAYGRADE DISTRIBUTION  
BY RATING

	PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9		PAYG1	PAYG2	PAYG3	PAYG4	PAYG5	PAYG6	PAYG7	PAYG8	PAYG9
7800 AN	.2928	.2399	.5007	.0000	.0000	.0000	.0000	.0000	.0000	4400 GS	.0000	.0595	.1191	.0595	.1786	.4048	.1429	.0357	.0000
6700 AB	.0305	.0366	.0586	.4664	.2137	.1429	.0427	.0037	.0073	8000 HT	.0677	.0737	.1139	.1848	.3211	.1616	.0628	.0180	.0052
6600 AC	.0435	.0543	.0688	.1015	.3587	.2609	.0942	.0145	.0109	4300 HM	.0058	.0082	.0360	.2893	.3171	.2218	.1039	.0089	.0096
6080 AF	.0183	.0461	.1299	.2455	.2770	.2042	.0557	.0194	.0055	2300 IS	.0104	.0026	.0207	.2163	.3420	.2928	.1075	.0039	.0039
7100 AG	.0026	.0131	.0471	.3194	.3115	.2173	.0681	.0131	.0078	2600 JO	.0000	.0000	.0270	.1460	.3405	.2811	.1676	.0270	.0108
7300 AK	.0213	.0164	.0716	.3366	.3279	.1489	.0551	.0164	.0077	3100 LI	.0000	.0000	.0328	.2705	.2869	.2869	.0574	.0328	.0328
6500 AO	.0184	.0684	.0806	.2735	.3214	.1602	.0694	.0082	.0020	1750 LN	.0000	.0000	.0000	.0000	.4959	.3333	.1382	.0244	.0081
7500 AS	.0000	.0174	.0291	.2820	.3924	.2093	.0465	.0233	.0000	0150 MA	.0000	.0000	.0000	.0000	.0115	.8702	.1031	.0153	.0000
6180 AV	.0264	.0429	.1142	.1881	.2992	.2564	.0515	.0193	.0061	4700 ML	.0164	.0328	.0000	.1147	.3771	.2951	.1147	.0492	.0000
6400 AW	.0307	.0347	.0935	.1108	.3391	.2377	.1255	.0147	.0174	3700 MM	.0013	.0119	.0882	.2120	.2145	.3076	.1194	.0322	.0129
7400 AZ	.0428	.0242	.0502	.3048	.3234	.2026	.0428	.0093	.0056	0900 MN	.0799	.0243	.0799	.0868	.2882	.2743	.1111	.0451	.0208
0100 BM	.0000	.0006	.0300	.3226	.2868	.1912	.1400	.0096	.0102	3900 MR	.0101	.0056	.0236	.2691	.3311	.2477	.0946	.0146	.0045
4000 BT	.0008	.0170	.1163	.3213	.2327	.1826	.1025	.0208	.0062	2200 MS	.0310	.0556	.0767	.3010	.2752	.2166	.0383	.0073	.0013
6000 CN	.4071	.0577	.5385	.0000	.0000	.0000	.0000	.0000	.0000	3300 MU	.0000	.0000	.0000	.0000	.5000	.5000	.0000	.0000	.0000
1622 CTA	.0230	.0268	.0690	.2529	.3755	.1954	.0498	.0115	.0000	1400 NC	.0000	.0000	.0000	.0000	.0000	.7500	.2500	.0000	.0000
1666 CTT	.0259	.0000	.0104	.1036	.4093	.3005	.1243	.0155	.0155	0300 OS	.0432	.0623	.1042	.2546	.2811	.1825	.0512	.0160	.0105
1633 CTM	.0296	.0000	.0074	.0259	.2815	.5519	.0778	.0111	.0185	0450 OT	.0000	.0000	.0180	.1351	.4324	.2432	.1171	.0450	.0090
1644 CTR	.0379	.0757	.1381	.1693	.3296	.1938	.0423	.0156	.0022	2700 PC	.0000	.0000	.0209	.4817	.2461	.1989	.0314	.0105	.0105
1655 CTR	.0307	.0552	.0900	.0941	.2986	.2699	.1452	.0143	.0061	7600 PH	.0025	.0025	.0546	.2010	.3474	.3176	.0521	.0074	.0149
1611 CTT	.0067	.0000	.0201	.1443	.4396	.2752	.0805	.0201	.0134	1080 PI	.0045	.0317	.0271	.1448	.3529	.2941	.1177	.0226	.0045
5080 CU	.0470	.0736	.1117	.2455	.2900	.1468	.0708	.0141	.0059	4600 PM	.0000	.0000	.0000	.1579	.4211	.2632	.1579	.0000	.0000
8300 DN	.0099	.0161	.0608	.3189	.3759	.1526	.0595	.0050	.0025	1800 PN	.0088	.0226	.0726	.1217	.4004	.2875	.0658	.0078	.0137
2100 DK	.0087	.0116	.0378	.2413	.4360	.2064	.0407	.0087	.0087	7800 PR	.0152	.0254	.0711	.2335	.3959	.2132	.0305	.0152	.0000
3200 DM	.0000	.0000	.0115	.1379	.4770	.2701	.0805	.0115	.0115	0200 QM	.0070	.0181	.0371	.1817	.3795	.2199	.1145	.0371	.0060
1900 DP	.0078	.0190	.0503	.2137	.3948	.2394	.0571	.0134	.0056	1500 RM	.0331	.0465	.0538	.2717	.3256	.1943	.0679	.0086	.0022
1010 DS	.0000	.0198	.0198	.0754	.3333	.3889	.1230	.0278	.0119	2500 RP	.1023	.0114	.0454	.2045	.4205	.2045	.0227	.0000	.0000
4100 EM	.0055	.0109	.0386	.2764	.2924	.2626	.0990	.0105	.0050	3600 SN	.2851	.2721	.4809	.0000	.0000	.0000	.0000	.0000	.0000
3800 EN	.0097	.0301	.0542	.3249	.3126	.1783	.0736	.0129	.0048	2490 SH	.0101	.0188	.1000	.4203	.2957	.1203	.0246	.0072	.0044
5380 EQ	.0452	.0885	.1202	.1343	.3681	.1750	.0573	.0119	.0041	2000 SK	.0233	.0264	.0521	.2528	.3316	.2097	.0714	.0262	.0098
1000 ET	.0142	.0066	.0323	.1329	.3234	.3451	.1091	.0288	.0096	0250 SM	.0354	.0224	.0633	.2067	.3855	.2365	.0410	.0093	.0037
0350 EW	.0328	.0697	.1844	.2008	.2254	.1721	.0943	.0082	.0164	0400 ST	.0286	.0708	.0934	.1657	.2726	.2199	.1099	.0271	.0151
5000 FN	.2216	.3459	.4654	.0000	.0000	.0000	.0000	.0000	.0000	7200 TD	.0000	.0000	.1400	.1600	.3600	.3000	.0400	.0000	.0000
0800 FT	.0346	.0432	.0653	.1854	.2517	.2344	.1153	.0404	.0346	0500 TM	.0176	.0193	.0580	.2285	.3111	.2091	.1388	.0070	.0123
0600 GM	.0257	.0203	.0603	.2822	.3407	.1694	.0782	.0215	.0054	5800 UT	.0474	.0620	.1169	.2195	.3198	.1650	.0541	.0162	.0047
										1700 YN	.0024	.0111	.0427	.2620	.3895	.2069	.0744	.0074	.0040

TABLE C-5  
ALLOCATION OF SAM AND OTHER SELRES  
TO RATING GROUPS

SAM ALLOCATIONS

RATING	%	RATING	%	RATING	%
7500 AN	.0890	6700 AB	.0047	6600 AC	.0044
6000 AF	.0116	7100 AG	.0008	7300 AK	.0068
6500 AO	.0039	7500 AS	.0003	6100 AV	.0207
6400 AW	.0040	7400 AZ	.0040	0100 BM	.0003
4000 BT	.0005	6000 CN	.0016	1622 CTA	.0010
1666 CTI	.0009	1633 CTM	.0013	1644 CTO	.0053
1655 CTR	.0043	1611 CTT	.0005	5000 CU	.0388
8300 DN	.0024	2100 DK	.0010	3200 DM	.0001
1900 DP	.0015	1010 DS	.0002	4100 EM	.0029
3800 EN	.0030	5380 EQ	.0451	1000 ET	.0055
0350 EW	.0021	5000 FN	.0939	0800 FT	.0070
0600 GM	.0078	4400 GS	.0000	8000 HM	.1058
4300 HT	.0044	2300 IS	.0021	2600 JO	.0000
3100 LI	.0000	1750 LN	.0000	0150 MA	.0000
4700 ML	.0003	3700 MM	.0006	0900 MN	.0040
3900 MR	.0021	2200 MS	.0163	3300 MJ	.0000
1400 NC	.0000	0300 OS	.0170	0450 OT	.0000
2700 PC	.0000	7600 PH	.0007	1000 PI	.0006
4600 PM	.0000	1800 PN	.0018	7000 PR	.0003
0200 QM	.0014	1500 RM	.0250	2500 RP	.0013
3600 SN	.3787	2490 SH	.0014	2000 SK	.0236
0250 SM	.0041	0400 ST	.0053	7200 TD	.0000
0500 TM	.0017	5800 UT	.0212	1700 YN	.0030

TABLE C-5 (Continued)

## OTHER ALLOCATIONS

RATING	%	RATING	%	RATING	%
7800 AN	.0255	6700 AB	.0107	6600 AC	.0033
6080 AF	.0412	7100 AG	.0037	7300 AK	.0133
6500 AO	.0071	7500 AS	.0039	6180 AV	.0349
6400 AW	.0043	7400 AZ	.0081	0100 BM	.0339
4000 BT	.0165	6000 CN	.0028	1622 CTA	.0038
1666 CTT	.0020	1633 CTM	.0026	1644 CTO	.0028
1655 CTR	.0037	1611 CTT	.0029	5080 CU	.0291
8300 DN	.0093	2100 DK	.0020	3200 DM	.0027
1900 DP	.0166	1010 DS	.0033	4100 EM	.0247
3800 EN	.0176	5380 EQ	.0315	1000 ET	.0207
0350 EW	.0022	5000 FN	.0243	0800 FT	.0080
0600 GM	.0141	4400 GS	.0014	8000 HM	.0726
4300 HT	.0333	2300 IS	.0152	2600 JO	.0032
3100 LI	.0006	1750 LN	.0028	0150 MA	.0096
4700 ML	.0005	3700 MM	.0114	0900 MN	.0014
3900 MR	.0129	2200 MS	.0303	3300 MJ	.0000
1400 NC	.0000	0300 OS	.0145	0450 OT	.0002
2700 PC	.0011	7600 PH	.0047	1080 PI	.0031
4600 PM	.0002	1800 PN	.0058	7000 PR	.0031
0200 QM	.0090	1500 RM	.0335	2500 RP	.0002
3600 SN	.1441	2490 SH	.0096	2000 SK	.0697
0250 SM	.0055	0400 ST	.0059	7200 TD	.0005
0500 TM	.0047	5800 UT	.0266	1700 YN	.0254



TABLE C-6

SELRES REQUIREMENTS BY  
FISCAL YEAR, RATING AND PAYGRADE

Requirements for 1986

PAY	7800 AN	6700 AB	6600 AL	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	3831.	21.	4.	724.	13.	89.	190.	33.	547.	106.	100.	211.	145.	113.
4	0.	167.	77.	852.	127.	296.	257.	109.	837.	189.	197.	1407.	241.	0.
5	0.	134.	73.	1113.	94.	367.	317.	166.	1067.	406.	148.	1038.	291.	0.
6	0.	145.	53.	681.	131.	148.	181.	76.	760.	201.	113.	868.	249.	0.
7	0.	38.	24.	263.	34.	56.	75.	17.	208.	105.	25.	487.	138.	0.
8	0.	8.	5.	92.	7.	20.	12.	10.	100.	23.	2.	53.	33.	0.
9	0.	4.	3.	36.	3.	9.	1.	1.	42.	14.	1.	23.	9.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	22.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4	80.	124.	91.	103.	26.	12.	1495.	6.	62.	0.	48.	1.	169.	165.
5	71.	169.	182.	164.	205.	77.	1488.	336.	58.	28.	208.	52.	507.	450.
6	27.	143.	74.	128.	255.	125.	1179.	269.	87.	63.	343.	82.	604.	624.
7	13.	53.	23.	58.	153.	86.	722.	115.	62.	31.	244.	72.	472.	315.
8	2.	2.	2.	13.	65.	30.	336.	49.	7.	17.	69.	32.	203.	142.
9	0.	0.	0.	3.	11.	5.	74.	5.	3.	1.	13.	7.	36.	18.
				0.	2.	4.	30.	2.	1.	1.	1.	3.	11.	14.

TABLE C-6 (Continued)

Requirements for 1986

PAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1628.	62.	25.	2149.	240.	199.	6.	826.	149.	45.	1.	19.	0.	0.
4	2276.	433.	53.	0.	219.	589.	32.	5502.	859.	134.	23.	26.	0.	0.
5	1988.	597.	31.	0.	195.	477.	54.	2593.	863.	347.	69.	24.	60.	30.
6	981.	496.	33.	0.	206.	364.	40.	1348.	636.	377.	40.	13.	63.	195.
7	350.	210.	33.	0.	98.	192.	27.	593.	328.	111.	33.	7.	16.	37.
8	85.	39.	1.	0.	32.	29.	3.	133.	31.	7.	0.	0.	4.	15.
9	29.	18.	1.	0.	26.	17.	0.	40.	29.	4.	1.	1.	1.	1.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	5.	59.	31.	35.	1032.	0.	0.	365.	2.	6.	14.	22.	0.	157.
4	18.	448.	105.	211.	758.	0.	0.	251.	20.	64.	83.	80.	21.	150.
5	28.	464.	174.	281.	729.	0.	0.	322.	47.	21.	152.	98.	14.	297.
6	19.	577.	85.	247.	597.	0.	10.	213.	52.	41.	108.	82.	15.	256.
7	5.	217.	38.	105.	109.	0.	0.	110.	25.	2.	25.	35.	3.	70.
8	3.	53.	14.	13.	30.	0.	0.	20.	0.	2.	2.	1.	0.	7.
9	0.	29.	8.	4.	2.	0.	0.	9.	0.	2.	0.	3.	0.	9.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	51.	47.	380.	63.	6783.	280.	183.	196.	182.	0.	35.	757.	316.	24588.
4	52.	190.	903.	113.	0.	174.	1920.	344.	84.	0.	237.	902.	970.	26868.
5	75.	412.	956.	37.	0.	176.	1527.	439.	86.	0.	240.	816.	1237.	25356.
6	42.	282.	633.	17.	0.	68.	1020.	147.	112.	0.	184.	463.	666.	16908.
7	3.	133.	250.	13.	0.	21.	502.	24.	39.	0.	80.	148.	303.	6817.
8	6.	43.	39.	2.	0.	13.	139.	3.	9.	0.	5.	61.	34.	1425.
9	1.	7.	4.	1.	0.	1.	44.	2.	8.	0.	7.	17.	13.	554.

TABLE C-6 (Continued)

Requirements for 1987

IPAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	4357.	21.	4.	800.	14.	89.	207.	34.	571.	107.	100.	289.	167.	157.
4	0.	179.	77.	892.	127.	307.	275.	111.	898.	181.	200.	1451.	289.	0.
5	0.	150.	75.	1194.	94.	369.	339.	173.	1113.	405.	151.	1054.	323.	0.
6	0.	158.	53.	776.	135.	159.	192.	78.	799.	208.	123.	871.	270.	0.
7	0.	46.	24.	283.	34.	64.	79.	17.	224.	107.	26.	491.	146.	0.
8	0.	10.	5.	104.	8.	20.	12.	12.	102.	24.	3.	55.	40.	0.
9	0.	4.	3.	36.	3.	9.	1.	1.	42.	15.	1.	42.	10.	0.

IPAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	33.	2.	0.	103.	26.	13.	1559.	13.	76.	0.	52.	1.	185.	166.
4	87.	124.	92.	164.	206.	81.	1575.	344.	78.	28.	209.	52.	543.	460.
5	79.	173.	186.	127.	254.	129.	1250.	280.	107.	63.	342.	82.	632.	636.
6	34.	145.	74.	59.	153.	83.	757.	121.	84.	46.	244.	72.	519.	326.
7	19.	53.	25.	13.	65.	27.	354.	51.	7.	17.	69.	32.	216.	149.
8	2.	2.	2.	3.	12.	6.	78.	5.	3.	1.	13.	7.	43.	18.
9	0.	0.	0.	0.	2.	4.	32.	2.	1.	1.	1.	3.	13.	14.

TABLE C-6 (Continued)

Requirements for 1987

IPAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	1200 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1675.	62.	28.	2410.	232.	203.	7.	941.	152.	47.	5.	22.	0.	0.
4	2352.	441.	63.	0.	220.	594.	32.	5883.	949.	149.	24.	26.	0.	0.
5	2099.	600.	36.	0.	196.	515.	54.	2899.	868.	407.	70.	26.	64.	30.
6	1037.	498.	38.	0.	205.	381.	40.	1492.	650.	388.	44.	15.	54.	196.
7	363.	230.	33.	0.	100.	200.	27.	646.	360.	111.	34.	7.	16.	40.
8	89.	43.	2.	0.	32.	50.	3.	164.	37.	7.	0.	0.	4.	16.
9	31.	21.	1.	0.	28.	17.	0.	50.	30.	4.	1.	1.	1.	1.

IPAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	8.	76.	34.	36.	1107.	0.	0.	411.	4.	7.	14.	22.	0.	171.
4	19.	469.	106.	234.	821.	0.	0.	270.	23.	79.	85.	80.	21.	178.
5	28.	487.	176.	283.	739.	0.	0.	339.	48.	28.	152.	98.	15.	355.
6	19.	600.	89.	261.	607.	0.	10.	222.	53.	57.	125.	85.	17.	298.
7	5.	247.	39.	108.	118.	0.	0.	110.	26.	3.	25.	35.	3.	81.
8	3.	84.	14.	16.	30.	0.	0.	22.	0.	2.	2.	1.	0.	10.
9	0.	29.	8.	4.	2.	0.	0.	15.	0.	2.	e.	3.	0.	9.

IPAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	51.	47.	376.	63.	6858.	283.	218.	195.	190.	0.	35.	861.	342.	26319.
4	63.	183.	904.	113.	0.	217.	1948.	343.	98.	0.	268.	934.	1016.	28235.
5	70.	412.	958.	40.	0.	179.	1581.	444.	96.	0.	261.	848.	1548.	26829.
6	47.	296.	641.	17.	0.	71.	1089.	148.	154.	0.	198.	489.	795.	17968.
7	3.	133.	259.	13.	0.	21.	503.	25.	43.	0.	91.	155.	318.	7169.
8	6.	43.	40.	2.	0.	13.	140.	3.	9.	0.	9.	65.	35.	1586.
9	1.	7.	8.	1.	0.	1.	44.	2.	8.	0.	7.	19.	17.	613.

TABLE C-6 (Continued)

Requirements for 1988

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	5117.	23.	5.	858.	17.	93.	217.	34.	575.	108.	103.	303.	182.	159.
4	0.	208.	78.	962.	129.	312.	279.	115.	941.	181.	209.	1462.	320.	0.
5	0.	172.	76.	1318.	95.	377.	342.	175.	1149.	408.	159.	1095.	353.	0.
6	0.	179.	56.	872.	139.	165.	195.	78.	859.	212.	124.	870.	304.	0.
7	0.	51.	26.	306.	35.	65.	80.	18.	229.	109.	27.	497.	164.	0.
8	0.	11.	6.	114.	10.	21.	13.	12.	106.	25.	3.	61.	43.	0.
9	0.	4.	4.	37.	3.	9.	1.	1.	43.	16.	1.	44.	11.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	33.	3.	0.	108.	27.	13.	1567.	16.	82.	0.	59.	3.	184.	166.
4	87.	127.	93.	166.	210.	81.	1586.	346.	90.	28.	213.	58.	578.	494.
5	80.	173.	186.	131.	256.	130.	1262.	288.	109.	65.	345.	86.	671.	676.
6	35.	145.	75.	60.	155.	84.	764.	122.	85.	48.	247.	75.	568.	354.
7	20.	54.	26.	14.	67.	27.	358.	52.	9.	18.	70.	34.	231.	160.
8	2.	2.	2.	3.	12.	6.	79.	5.	4.	2.	15.	8.	46.	18.
9	0.	0.	0.	0.	2.	4.	32.	2.	1.	1.	1.	4.	13.	15.

TABLE C-6 (Continued)

Requirements for 1988

PAY	15380 EQ	1000 ET	03350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	3.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1683.	64.	30.	2509.	237.	213.	11.	1055.	154.	49.	7.	23.	0.	0.
4	2361.	466.	79.	0.	224.	605.	38.	6559.	1033.	151.	26.	27.	0.	0.
5	2112.	622.	37.	0.	206.	529.	62.	3225.	896.	409.	73.	27.	64.	34.
6	1042.	512.	40.	0.	214.	387.	42.	1655.	676.	389.	46.	17.	54.	200.
7	367.	237.	35.	0.	102.	210.	27.	713.	370.	114.	35.	7.	17.	46.
8	90.	44.	3.	0.	33.	51.	4.	168.	40.	8.	0.	0.	5.	21.
9	31.	21.	1.	0.	29.	19.	0.	53.	30.	4.	1.	1.	1.	2.

PAY	14700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	9.	90.	36.	37.	1109.	0.	0.	436.	6.	8.	15.	24.	0.	178.
4	21.	564.	108.	238.	831.	0.	0.	332.	25.	88.	85.	83.	22.	201.
5	29.	566.	178.	295.	759.	0.	0.	358.	49.	30.	156.	104.	15.	371.
6	22.	684.	91.	262.	609.	0.	10.	247.	56.	59.	132.	90.	17.	315.
7	6.	281.	40.	111.	119.	0.	0.	113.	27.	3.	27.	36.	3.	83.
8	3.	92.	15.	16.	31.	0.	0.	24.	0.	2.	3.	2.	0.	10.
9	0.	30.	9.	5.	2.	0.	0.	15.	0.	2.	0.	3.	0.	9.

PAY	17000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	54.	47.	380.	65.	7008.	284.	240.	195.	211.	0.	35.	869.	310.	27736.
4	67.	184.	923.	117.	0.	245.	1950.	343.	134.	0.	271.	941.	1071.	29766.
5	72.	416.	959.	42.	0.	200.	1583.	453.	103.	0.	268.	853.	1612.	27944.
6	54.	303.	643.	19.	0.	72.	1097.	152.	163.	0.	199.	494.	831.	18765.
7	4.	134.	261.	14.	0.	23.	504.	25.	45.	0.	91.	159.	335.	7471.
8	7.	43.	42.	2.	0.	14.	142.	3.	11.	0.	9.	65.	37.	1669.
9	1.	8.	10.	1.	0.	1.	44.	3.	8.	0.	9.	19.	19.	640.

TABLE C-6 (Continued)

Requirements for 1989

IPAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	5000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	5285.	25.	6.	866.	18.	93.	220.	36.	582.	110.	104.	324.	189.	162.
4	0.	213.	79.	963.	132.	313.	282.	117.	949.	181.	208.	1479.	323.	0.
5	0.	175.	77.	1323.	96.	378.	344.	177.	1155.	409.	161.	1110.	356.	0.
6	0.	183.	57.	884.	144.	169.	199.	83.	852.	214.	127.	870.	307.	0.
7	0.	54.	27.	309.	36.	66.	81.	19.	231.	110.	27.	503.	165.	0.
8	0.	11.	6.	117.	10.	22.	13.	13.	108.	26.	4.	65.	44.	0.
9	0.	5.	4.	38.	4.	10.	1.	1.	44.	16.	1.	44.	12.	0.

IPAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	35.	3.	0.	109.	31.	14.	1577.	19.	85.	0.	60.	3.	191.	174.
4	89.	129.	94.	167.	212.	82.	1593.	349.	91.	29.	215.	59.	604.	521.
5	82.	174.	187.	132.	259.	131.	1269.	291.	110.	66.	350.	87.	691.	684.
6	38.	145.	76.	60.	157.	85.	770.	124.	86.	50.	253.	82.	577.	362.
7	23.	54.	27.	15.	68.	29.	362.	53.	9.	19.	71.	35.	238.	164.
8	3.	3.	3.	4.	13.	7.	81.	5.	4.	2.	15.	8.	48.	18.
9	0.	0.	0.	0.	3.	5.	33.	2.	2.	2.	1.	4.	14.	15.

TABLE C-6 (Continued)

Requirements for 1989

PAY	5380	EQ	1000	ET	0350	EW	5000	FN	0800	FT	0600	GM	4400	GS	8000	HM	4300	HT	2300	IS	2600	JO	3100	LI	1750	LN	0150	MA
1		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.
2		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.		0.
3	1690.		64.		33.		2514.		237.		219.		14.		1065.		157.		50.		9.		25.		0.		0.	
4	2367.		468.		83.		0.		227.		610.		41.		6571.		1043.		153.		28.		29.		0.		0.	
5	2119.		623.		40.		0.		210.		540.		68.		3238.		900.		411.		77.		28.		65.		37.	
6	1051.		517.		42.		0.		216.		398.		44.		1666.		680.		391.		47.		18.		55.		205.	
7	370.		239.		36.		0.		103.		215.		29.		722.		374.		116.		36.		7.		18.		51.	
8	92.		44.		3.		0.		35.		53.		5.		171.		41.		8.		0.		0.		5.		26.	
9	32.		21.		2.		0.		29.		20.		0.		55.		31.		5.		2.		2.		2.		2.	

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	10.	93.	37.	38.	1117.	0.	0.	437.	7.	9.	16.	24.	0.	0.
4	22.	573.	111.	241.	837.	0.	0.	340.	26.	92.	86.	86.	23.	186.
5	30.	582.	179.	299.	768.	0.	0.	361.	50.	32.	158.	108.	16.	212.
6	26.	700.	93.	267.	615.	0.	13.	252.	56.	61.	134.	96.	19.	378.
7	6.	289.	41.	113.	121.	0.	0.	114.	28.	4.	25.	38.	4.	320.
8	4.	96.	15.	16.	32.	0.	0.	25.	0.	3.	4.	2.	0.	85.
9	0.	30.	9.	7.	3.	0.	0.	16.	0.	2.	0.	4.	0.	10.
														9.

PAY	7000	PR	0200	QM	1500	RM	2500	RP	3600	SN	2490	SH	2000	SK	0250	SM	0400	ST	7200	TD	0500	TM	5800	UT	1700	YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	56.	0.	49.	0.	388.	0.	67.	0.	7109.	0.	290.	0.	241.	0.	195.	0.	214.	0.	0.	0.	36.	0.	875.	0.	310.	0.	28202.
4	67.	0.	188.	0.	924.	0.	119.	0.	0.	0.	247.	0.	1957.	0.	345.	0.	135.	0.	0.	0.	274.	0.	947.	0.	1105.	0.	30050.
5	74.	0.	416.	0.	962.	0.	43.	0.	0.	0.	203.	0.	1585.	0.	459.	0.	106.	0.	0.	0.	271.	0.	857.	0.	1631.	0.	28198.
6	55.	0.	309.	0.	650.	0.	20.	0.	0.	0.	74.	0.	1101.	0.	156.	0.	168.	0.	0.	0.	201.	0.	501.	0.	839.	0.	19010.
7	4.	0.	137.	0.	262.	0.	15.	0.	0.	0.	24.	0.	505.	0.	25.	0.	48.	0.	0.	0.	94.	0.	163.	0.	347.	0.	7603.
8	7.	0.	44.	0.	43.	0.	3.	0.	0.	0.	14.	0.	143.	0.	4.	0.	11.	0.	0.	0.	11.	0.	67.	0.	40.	0.	1730.
9	1.	0.	8.	0.	12.	0.	2.	0.	0.	0.	2.	0.	45.	0.	3.	0.	8.	0.	0.	0.	11.	0.	20.	0.	23.	0.	679.



TABLE C-6 (Continued)

Requirements for 1990

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	5335.	27.	7.	874.	19.	94.	223.	38.	592.	111.	105.	331.	193.	170.
4	0.	218.	81.	975.	134.	314.	286.	121.	959.	183.	211.	1494.	325.	0.
5	0.	180.	78.	1331.	98.	381.	345.	181.	1162.	411.	162.	1120.	362.	0.
6	0.	187.	59.	894.	147.	170.	200.	84.	859.	216.	130.	873.	309.	0.
7	0.	55.	29.	311.	38.	67.	82.	21.	235.	111.	28.	507.	166.	0.
8	0.	12.	7.	118.	11.	23.	14.	13.	111.	27.	4.	67.	45.	0.
9	0.	0.	4.	39.	4.	10.	1.	1.	45.	16.	1.	45.	13.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	18300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	36.	3.	0.	110.	31.	15.	1589.	21.	86.	0.	61.	4.	197.	179.
4	90.	130.	95.	168.	213.	83.	1603.	354.	93.	31.	217.	60.	614.	537.
5	83.	175.	189.	133.	260.	132.	1274.	295.	111.	67.	355.	88.	704.	689.
6	38.	146.	77.	60.	158.	86.	774.	126.	87.	51.	256.	83.	585.	368.
7	24.	55.	28.	16.	69.	29.	364.	54.	10.	20.	73.	36.	243.	168.
8	3.	3.	3.	4.	13.	8.	83.	6.	4.	3.	16.	8.	49.	18.
9	0.	0.	0.	0.	3.	6.	34.	3.	2.	2.	1.	4.	14.	16.

TABLE C-6 (Continued)

Requirements for 1990

PAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1698.	65.	35.	2541.	237.	223.	17.	1080.	160.	52.	11.	26.	0.	0.
4	2374.	471.	84.	0.	230.	617.	44.	6586.	1053.	154.	30.	29.	0.	0.
5	2128.	624.	41.	0.	213.	546.	72.	3249.	904.	413.	78.	29.	67.	39.
6	1062.	519.	43.	0.	219.	404.	46.	1674.	684.	393.	48.	20.	56.	207.
7	374.	241.	37.	0.	106.	218.	30.	730.	378.	118.	37.	8.	20.	53.
8	93.	45.	3.	0.	37.	54.	6.	176.	42.	9.	0.	0.	6.	28.
9	32.	22.	3.	0.	29.	23.	0.	58.	31.	6.	2.	2.	3.	3.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	11.	97.	38.	39.	1122.	0.	0.	439.	8.	10.	17.	24.	0.	193.
4	23.	578.	112.	244.	772.	0.	0.	343.	28.	95.	88.	89.	24.	223.
5	31.	592.	181.	302.	619.	0.	0.	364.	52.	34.	160.	110.	17.	385.
6	27.	716.	94.	273.	123.	0.	16.	254.	58.	63.	137.	100.	19.	326.
7	7.	298.	42.	116.	33.	0.	0.	115.	29.	5.	26.	39.	5.	98.
8	4.	100.	16.	17.	4.	0.	0.	26.	0.	3.	4.	2.	0.	11.
9	0.	30.	9.	7.	4.	0.	0.	16.	0.	2.	0.	4.	0.	9.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	57.	50.	390.	69.	7241.	294.	243.	197.	222.	0.	38.	882.	315.	28592.
4	69.	191.	926.	121.	0.	249.	1976.	345.	137.	0.	275.	954.	1120.	30313.
5	75.	417.	964.	44.	0.	206.	1597.	462.	107.	0.	275.	863.	1661.	28440.
6	56.	313.	654.	21.	0.	75.	1108.	158.	171.	0.	203.	505.	872.	19236.
7	5.	139.	263.	16.	0.	25.	510.	25.	49.	0.	97.	166.	357.	7734.
8	7.	44.	44.	4.	0.	14.	145.	5.	11.	0.	12.	69.	45.	1788.
9	1.	8.	12.	2.	0.	3.	46.	3.	8.	0.	12.	20.	27.	706.

TABLE C-6 (Continued)

Requirements for 1991

PAY	7800 AN	6700 AB	6600 AC	6080 AF	7100 AG	7300 AK	6500 AO	7500 AS	6180 AV	6400 AW	7400 AZ	0100 BM	4000 BT	6000 CN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	5340.	27.	7.	876.	19.	95.	223.	38.	592.	111.	106.	332.	193.	170.
4	0.	219.	81.	976.	134.	314.	287.	121.	959.	183.	212.	1494.	325.	0.
5	0.	180.	78.	1333.	98.	381.	346.	153.	1164.	411.	162.	1120.	362.	0.
6	0.	188.	60.	895.	147.	171.	200.	84.	860.	216.	130.	873.	309.	0.
7	0.	57.	29.	314.	38.	67.	82.	21.	235.	111.	28.	507.	166.	0.
8	0.	12.	7.	119.	11.	23.	14.	14.	113.	27.	4.	69.	45.	0.
9	0.	5.	5.	39.	4.	10.	1.	1.	45.	16.	1.	46.	13.	0.

PAY	1622 CTA	1666 CTI	1633 CTM	1644 CTO	1655 CTR	1611 CTT	5080 CU	8300 DN	2100 DK	3200 DM	1900 DP	1010 DS	4100 EM	3800 EN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	36.	3.	0.	110.	31.	16.	1589.	21.	86.	0.	61.	4.	197.	179.
4	90.	130.	95.	168.	213.	83.	1604.	354.	93.	31.	217.	60.	615.	539.
5	83.	175.	189.	133.	260.	133.	1275.	295.	111.	67.	355.	88.	704.	689.
6	38.	146.	77.	60.	158.	86.	775.	127.	87.	51.	256.	83.	585.	368.
7	24.	55.	28.	16.	69.	30.	365.	54.	10.	20.	73.	36.	243.	168.
8	3.	3.	3.	4.	14.	8.	84.	6.	5.	3.	16.	8.	50.	18.
9	0.	0.	0.	0.	3.	6.	34.	3.	2.	2.	1.	4.	15.	16.

TABLE C-6 (Continued)

Requirements for 1991

PAY	5380 EQ	1000 ET	0350 EW	5000 FN	0800 FT	0600 GM	4400 GS	8000 HM	4300 HT	2300 IS	2600 JO	3100 LI	1750 LN	0150 MA
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	1699.	65.	35.	2519.	237.	243.	17.	1081.	160.	52.	11.	26.	0.	0.
4	2374.	471.	84.	0.	230.	617.	44.	6587.	1054.	154.	30.	29.	0.	0.
5	2129.	624.	42.	0.	213.	546.	73.	3249.	904.	413.	78.	29.	67.	39.
6	1062.	519.	43.	0.	219.	404.	46.	1675.	685.	393.	48.	20.	57.	207.
7	374.	241.	38.	0.	106.	218.	31.	731.	378.	119.	37.	8.	20.	54.
8	94.	45.	4.	0.	37.	54.	6.	176.	42.	9.	0.	0.	6.	29.
9	33.	22.	3.	0.	29.	23.	0.	59.	32.	6.	3.	2.	3.	3.

PAY	4700 ML	3700 MM	0900 MN	3900 MR	2200 MS	3300 MU	1400 NC	0300 OS	0450 OT	2700 PC	7600 PH	1080 PI	4600 PM	1800 PN
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	11.	98.	38.	39.	1122.	0.	0.	439.	8.	11.	17.	24.	0.	196.
4	23.	579.	112.	244.	842.	0.	0.	343.	28.	95.	89.	89.	24.	223.
5	31.	594.	181.	302.	772.	0.	0.	372.	52.	34.	161.	110.	17.	385.
6	27.	716.	94.	273.	619.	0.	16.	254.	58.	63.	137.	100.	19.	326.
7	7.	298.	42.	116.	125.	0.	0.	115.	29.	5.	26.	39.	5.	88.
8	5.	100.	16.	17.	33.	0.	0.	26.	0.	4.	5.	2.	0.	11.
9	0.	31.	9.	7.	4.	0.	0.	17.	0.	2.	0.	4.	0.	9.

PAY	7000 PR	0200 QM	1500 RM	2500 RP	3600 SN	2490 SH	2000 SK	0250 SM	0400 ST	7200 TD	0500 TM	5800 UT	1700 YN	TOTAL
1	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3	57.	50.	390.	69.	7242.	294.	243.	197.	222.	0.	0.	883.	318.	28593.
4	69.	191.	926.	121.	0.	249.	1976.	345.	138.	0.	38.	955.	1121.	30328.
5	75.	417.	964.	44.	0.	206.	1597.	462.	107.	0.	275.	863.	1661.	28463.
6	56.	313.	654.	21.	0.	76.	1108.	158.	171.	0.	203.	507.	872.	19249.
7	5.	135.	263.	16.	0.	25.	510.	25.	49.	0.	97.	166.	357.	7748.
8	7.	44.	44.	4.	0.	14.	145.	5.	12.	0.	12.	69.	48.	1808.
9	1.	9.	12.	3.	0.	3.	46.	4.	8.	0.	12.	20.	30.	721.

ANNEX C-1

PROGRAM LISTINGS FOR COMPUTING  
SELRES AFFILIATION AND CONTINUATION RATES

```

PROGRAM MATCH_LOSS

IMPLICIT NONE

! LOSS-FILE VARIABLES
STRUCTURE /LOSS_FILE/
  INTEGER SSN, STR_LOSS, LOSS_REASON
  CHARACTER*2 RQC
  INTEGER PROGRAM, RATING, PAYGRADE, LOS
END STRUCTURE
RECORD /LOSS_FILE/ CURR, NEXT
INTEGER EOF_A, EOF_N

INTEGER T_TYPE_T
INTEGER FY, KRATE, PG
INTEGER LOSS_YR_A, LOSS_YR_N
INTEGER T_TYPE, R_TYPE
COMMON /SUBSCRIPT/CURR, NEXT, LOSS_YR_A, LOSS_YR_N, T_TYPE, R_TYPE
EXTERNAL HANDLER

INTEGER*4 LOSS(6,109), GAIN(6,109,0:10,11,3), GAIN_AF(6,109)
REAL      RATE(6)

INTEGER*2 I, J, LOS, L, M, N, P, IO, ST
INTEGER   UP, UPFLAG, GAIN_CNT, LOSS_CNT
INTEGER   TEMAC, TEMAC_NO_MATCH, OLD_MATCH, OLD_NO_MATCH
INTEGER   T_TYPE_CNT(3), R_TYPE_CNT(2), PREV_CNT, CNT_943, GP
INTEGER   INCNT1, INCNT2
INTEGER   AA, BB
LOGICAL   PREV
INTEGER   GAIN_TOT(2), LOSS_TOT(2)

CHARACTER*3 RATE_NAME(109)
COMMON /KEVIN_RATING/RATE_NAME

! IEMF-LONGIT VARIABLES
STRUCTURE /LONGIT_FILE/
  INTEGER SSN, YRFLAG(11), RATING(11), PAYGRADE(11)
END STRUCTURE
RECORD /LONGIT_FILE/ LONG
INTEGER EOF_I

CALL LIB$ESTABLISH(HANDLER)

CALL KEVIN_TO_RATING(0,0,0,0)

CALL READ_LOSSES(CURR, EOF_A)
  INCNT1=INCNT1+1
LOSS_YR_A=FY(CURR.STR_LOSS)
DO WHILE (LOSS_YR_A.LT.4)
  CALL READ_LOSSES(CURR, EOF_A)
  INCNT1=INCNT1+1
  LOSS_YR_A=FY(CURR.STR_LOSS)
END DO

```

```

CALL READ_LOSSES(NEXT, EOF_N)
  INCNT1=INCNT1+1
LOSS_YR_N=FY(NEXT.STR_LOSS)
DO WHILE (LOSS_YR_N.LT.4)
  CALL READ_LOSSES(NEXT, EOF_N)
  INCNT1=INCNT1+1
  LOSS_YR_N=FY(NEXT.STR_LOSS)
END DO

CALL READ_LONGIT(LONG, EOF_I)
  INCNT2=INCNT2+1
AA=0
BB=0
WRITE(6,*) '          GAINS          LOSSES          TEMAC '

DO WHILE (EOF_A.EQ.0)

  IF (AA.EQ.10000) THEN
    AA=0
    BB=BB+1
    WRITE(6,*) BB, GAIN_CNT, LOSS_CNT, TEMAC
  END IF
  AA=AA+1

  IF (CURR.SSN.EQ.LONG.SSN) THEN
    if not TEMAC
    IF (LOSS_YR_A.EQ.1) THEN
      PREV=.TRUE.
    ELSE
      PREV= (LONG.YRFLAG(LOSS_YR_A-1).NE.1)
    END IF
    IF (((CURR.LOSS_REASON.NE.943).OR.(CURR.PROGRAM.EQ.2))
    * .AND. PREV ) THEN
      IF (LOSS_YR_A.EQ.10) THEN
        IF ((CURR.PROGRAM.EQ.1).OR.(CURR.PROGRAM.EQ.3).OR.
        * (CURR.PROGRAM.EQ.4).OR.(CURR.PROGRAM.EQ.5).OR.
        * (CURR.PROGRAM.EQ.6)) THEN
          LOSS(1,CURR.RATING)=LOSS(1,CURR.RATING)+1
          IF (CURR.PROGRAM.EQ.1) THEN
            LOSS(3,CURR.RATING)=LOSS(3,CURR.RATING)+1
          ELSE
            IF (CURR.PROGRAM.EQ.3)
            * LOSS(4,CURR.RATING)=LOSS(4,CURR.RATING)+1
          END IF
        ELSE
          LOSS(2,CURR.RATING)=LOSS(2,CURR.RATING)+1
        END IF
        IF (CURR.LOS.LE.6) THEN
          LOSS(5,CURR.RATING)=LOSS(5,CURR.RATING)+1
        ELSE
          LOSS(6,CURR.RATING)=LOSS(6,CURR.RATING)+1
        END IF
        LOSS_CNT=LOSS_CNT+1
      ELSE
        OLD_MATCH=OLD_MATCH+1
      END IF
    END IF
  END IF

```

```

IF (CURR.SSN.EQ.NEXT.SSN) THEN
  UP=LOSS_YR_N-1
  UPFLAG=1
ELSE
  UP=11
  UPFLAG=0
END IF

DO I= LOSS_YR_A,UP
  IF (LONG.YRFLAG(I).EQ.1) THEN
    CALL TYPE_GAIN(CURR, LONG, LOSS_YR_A, I, UP,
      *      T_TYPE, R_TYPE, KRATE, PG, LOSS_CNT)
    IF (T_TYPE.LE.2) THEN
      T_TYPE_T=T_TYPE
      J=I
      DO WHILE ((J.LE.1).AND.(LONG.YRFLAG(J).EQ.1))
        LOS=J-I
        CALL TYPE_GAIN(CURR, LONG, LOSS_YR_A, J, UP,
          *      T_TYPE, R_TYPE, KRATE, PG, LOSS_CNT) !get "current" rating
        IF ((CURR.PROGRAM.EQ.1).OR.(CURR.PROGRAM.EQ.3).OR.
          *      (CURR.PROGRAM.EQ.4).OR.(CURR.PROGRAM.EQ.5).OR.
          *      (CURR.PROGRAM.EQ.6)) THEN
          GAIN(1, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
          *      GAIN(1, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
          IF (CURR.PROGRAM.EQ.1) THEN
            GAIN(3, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
            *      GAIN(3, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
          ELSE
            IF (CURR.PROGRAM.EQ.3)
              *      GAIN(4, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
              *      GAIN(4, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
            END IF
          ELSE
            GAIN(2, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
            *      GAIN(2, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
          END IF
          IF (CURR.LOS.LE.6) THEN
            GAIN(5, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
            *      GAIN(5, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
          ELSE
            GAIN(6, KRATE, LOS, LOSS_YR_A, T_TYPE_T)=
            *      GAIN(6, KRATE, LOS, LOSS_YR_A, T_TYPE_T)+1
          END IF
          J=J+1
        END DO
      END IF
      IF (PG.NE. CURR.PAYGRADE) GP=GP+1
      GAIN_CNT=GAIN_CNT+1
      T_TYPE_CNT(T_TYPE)=T_TYPE_CNT(T_TYPE)+1
      R_TYPE_CNT(R_TYPE)=R_TYPE_CNT(R_TYPE)+1
      GOTO 100
    END IF ! YRFLAG MATCH?
  END DO ! DO ALL POSSIBLE GOOD YRFLAGS
  CONTINUE
ELSE

```

100



```

    TEMAC=TEMAC+1
    IF (.NOT.PREV) THEN
        PREV_CNT=PREV_CNT+1
    ELSE
        CNT_943=CNT_943+1
    END IF
END IF

```

```

CURR=NEXT
EOF_A=EOF_N
LOSS_YR_A=LOSS_YR_N
CALL READ_LOSSES(NEXT, EOF_N)
INCNT1=INCNT1+1
LOSS_YR_N=FY(NEXT.STR_LOSS)
DO WHILE (LOSS_YR_N.LT.4)
    CALL READ_LOSSES(NEXT, EOF_N)
    INCNT1=INCNT1+1
    LOSS_YR_N=FY(NEXT.STR_LOSS)
END DO

```

```

ELSE
    IF (CURR.SSN.LT. LONG.SSN) THEN
        if not TEMAC
        IF (CURR.LOSS_REASON.NE.943) THEN
            IF (LOSS_YR_A.EQ.10) THEN
                IF ((CURR.PROGRAM.EQ.1).OR.(CURR.PROGRAM.EQ.3).OR.
                    (CURR.PROGRAM.EQ.4).OR.(CURR.PROGRAM.EQ.5).OR.
                    (CURR.PROGRAM.EQ.6)) THEN
                    * LOSS(1,CURR.RATING)=LOSS(1,CURR.RATING)+1
                    * IF (CURR.PROGRAM.EQ.1) THEN
                        LOSS(3,CURR.RATING)=LOSS(3,CURR.RATING)+1
                    ELSE
                        IF (CURR.PROGRAM.EQ.3)
                        * LOSS(4,CURR.RATING)=LOSS(4,CURR.RATING)+1
                        END IF
                    ELSE
                        LOSS(2,CURR.RATING)=LOSS(2,CURR.RATING)+1
                    END IF
                    IF (CURR.LOS.LE.6) THEN
                        LOSS(5,CURR.RATING)=LOSS(5,CURR.RATING)+1
                    ELSE
                        LOSS(6,CURR.RATING)=LOSS(6,CURR.RATING)+1
                    END IF
                    LOSS_CNT=LOSS_CNT+1
                ELSE
                    OLD_NO_MATCH=OLD_NO_MATCH+1
                END IF
            ELSE
                TEMAC_NO_MATCH=TEMAC_NO_MATCH+1
            END IF ! TEMAC
        CURR=NEXT
        EOF_A=EOF_N
        LOSS_YR_A=LOSS_YR_N
        CALL READ_LOSSES(NEXT, EOF_N)
        INCNT1=INCNT1+1
        LOSS_YR_N=FY(NEXT.STR_LOSS)
    
```

```

      DO WHILE (LOSS_YR_N.LT.4)
        CALL READ_LOSSES(NEXT,EOF_N)
        INCNT1=INCNT1+1
        LOSS_YR_N=FY(NEXT.STR_LOSS)
      END DO
    ELSE
      CALL READ_LONGIT(LONG,EOF_I)
      INCNT2=INCNT2+1
    END IF
  END IF
END DO

WRITE(6,*)
  WRITE(6,*) 'LOSS FILE RECORDS READ= ',INCNT1
  WRITE(6,*) 'LONGIT FILE RECORDS READ= ',INCNT2
  WRITE(6,*) 'MATCHING SSNs W/ LOSS PRIOR TO 84 ',OLD_MATCH
  WRITE(6,*) 'NON-MATCHING SSNs W/ LOSS PRIOR TO 84 ',OLD_NO_MATCH
  WRITE(6,*) ' GAIN_CNT= ',GAIN_CNT,' LOSS_CNT= ',LOSS_CNT
  WRITE(6,*) ' TEMACS= ',TEMAC,' TEMAC BY PREV= ',PREV_CNT,
*      ' TEMAC BY 943= ',CNT_943
  WRITE(6,*) ' NON-MATCHING TEMAC ',TEMAC_NO_MATCH
  WRITE(6,*) 'T_TYPE_CNT: ',(T_TYPE_CNT(I),I=1,3)
  WRITE(6,*) 'R_TYPE_CNT: ',(R_TYPE_CNT(I),I=1,2)
  WRITE(6,*) 'GAINS W/ DIFF PG',GP
  WRITE(6,*)

INCLUDE 'WRITE_MATCH3 /LIST'

END

```

C WRITE\_MATCH3.FOR

```
DO J=1,69
  DO I=1,6
    GAIN_AF(I,J)=GAIN(I,J,0,10,1)+GAIN(I,J,0,10,2)
  END DO
END DO

WRITE(7,*) '          TABLE OF SELRES AFFILIATION'
WRITE(7,*) '          RATING BY PROGRAM'
WRITE(7,*) '          4YO,5/6YO,PS,TAR          A/M'
DO J=1,69
  CALL WRITE_LINE(RATE_NAME(J),GAIN_AF(1,J),GAIN_AF(2,J),
*               LOSS(1,J),LOSS(2,J))
  DO I=1,2
    GAIN_TOT(I)=GAIN_TOT(I)+GAIN_AF(I,J)
    LOSS_TOT(I)=LOSS_TOT(I)+LOSS(I,J)
  END DO
END DO
WRITE(7,*) 'total',(GAIN_TOT(I),LOSS_TOT(I),
*             REAL(GAIN_TOT(I))/REAL(MAX(LOSS_TOT(I),1)),I=1,2)
DO I=1,2
  GAIN_TOT(I)=0
  LOSS_TOT(I)=0
END DO

WRITE(7,*)
WRITE(7,*) '          TABLE OF SELRES AFFILIATION'
WRITE(7,*) '          RATING BY PROGRAM'
WRITE(7,*) '          4YO          5/6YO '
DO J=1,69
  CALL WRITE_LINE(RATE_NAME(J),GAIN_AF(3,J),GAIN_AF(4,J),
*               LOSS(3,J),LOSS(4,J))
  DO I=1,2
    GAIN_TOT(I)=GAIN_TOT(I)+GAIN_AF(I+2,J)
    LOSS_TOT(I)=LOSS_TOT(I)+LOSS(I+2,J)
  END DO
END DO
WRITE(7,*) 'total',(GAIN_TOT(I),LOSS_TOT(I),
*             REAL(GAIN_TOT(I))/REAL(MAX(LOSS_TOT(I),1)),I=1,2)
DO I=1,2
  GAIN_TOT(I)=0
  LOSS_TOT(I)=0
END DO
```

```

WRITE(7,*)
WRITE(7,*) '
WRITE(7,*) '
WRITE(7,*)
WRITE(7,*) '
WRITE(7,*) '
DO J=1,69
  CALL WRITE_LINE(RATE_NAME(J),GAIN_AF(5,J),GAIN_AF(6,J),
*              LOSS(5,J),LOSS(6,J))
  DO I=1,2
    GAIN_TOT(I)=GAIN_TOT(I)+GAIN_AF(I+4,J)
    LOSS_TOT(I)=LOSS_TOT(I)+LOSS(I+4,J)
  END DO
END DO
WRITE(7,*) 'total',(GAIN_TOT(I),LOSS_TOT(I),
*      REAL(GAIN_TOT(I))/REAL(MAX(LOSS_TOT(I),1)),I=1,2)

DO T_TYPE=1,2
  DO L=1,11
    DO LOS=0,10
      DO J=1,69
        WRITE(8,*) (GAIN(I,J,LOS,L,T_TYPE),I=1,6)
      END DO
    END DO
  END DO
END DO
END DO

```

SUBROUTINE READ\_LOSSES(IN, EOF)

IMPLICIT NONE

STRUCTURE /LOSS\_FILE/

INTEGER SSN, STR\_LOSS, LOSS\_REASON

CHARACTER\*2 RQC

INTEGER PROGRAM, RATING, PAYGRADE, LOS

END STRUCTURE

RECORD /LOSS\_FILE/ IN

INCLUDE '(\$FORIOSDEF)/NOLIST'

INTEGER EOF, TMP

INTEGER IERR

0 CONTINUE

READ(1, 100, END=900, ERR=990, IOSTAT=IERR)

\* IN.SSN, IN.STR\_LOSS, IN.LOSS\_REASON, IN.RQC,

\* IN.PROGRAM, IN.RATING, IN.PAYGRADE, IN.LOS

00 FORMAT(I9, I4, I3, A2, X, I2, 3I3)

IF (IN.LOS .GE. 21) THEN

IN.LOS=21

ELSE

IF (IN.LOS .LE. 0) THEN

WRITE (6, \*) 'READ LOSS>> OLD LOS = ', IN.LOS, ' CONVERTED TO 1'

IN.LOS=1

END IF

END IF

CALL KEVIN\_TO\_RATING(3, IN.RATING, TMP, 0)

IN.RATING=TMP

RETURN

900 CONTINUE

IN.SSN=999999999

EOF=1

RETURN

990 CONTINUE

IF (IERR.EQ. FOR\$IOS\_INPCONERR) THEN

WRITE(6, \*) 'READ\_LOSS>> BAD RECORD: ',

\* IN.SSN, IN.STR\_LOSS, IN.LOSS\_REASON, IN.RQC,

\* IN.PROGRAM, IN.RATING, IN.PAYGRADE, IN.LOS

GOTO 10

END IF

WRITE(6, \*) 'READ\_LOSS>> INPUT ERROR:', IERR

STOP

END

```

SUBROUTINE READ_LONGIT(IN,EOF)

IMPLICIT NONE

STRUCTURE /LONGIT_FILE/
  INTEGER SSN,YRFLAG(11),RATING(11),PAYGRADE(11)
END STRUCTURE
RECORD /LONGIT_FILE/ IN

INCLUDE '($FORIOSDEF)/NOLIST'

INTEGER EOF
INTEGER IERR,T

10  CONTINUE
   READ(2,100,END=900,ERR=990,IOSTAT=IERR)
*   IN.SSN,(IN.YRFLAG(T),T=1,11),(IN.RATING(T),IN.PAYGRADE(T),T=1,11)
100  FORMAT(I9,11I1,11(I4,I1,X))

   RETURN

900  CONTINUE

   IN.SSN=999999999
   EOF=1

   RETURN

990  CONTINUE
   IF (IERR.EQ. FOR$IOS_INPCONERR) THEN
     WRITE(6,*) 'READ_LONGIT>> BAD RECORD: ',
*     IN.SSN,(IN.YRFLAG(T),T=1,11),
*     (IN.RATING(T),IN.PAYGRADE(T),T=1,11)
     GOTO 10
   END IF
   WRITE(6,*) 'READ_LONG>> INPUT ERROR:',IERR
   STOP

END

```

INTEGER FUNCTION FY(YMM)

IMPLICIT NONE

INTEGER YMM,YY,MM

YY=INT(YMM/100)

MM=YMM-YY\*100

IF (MM.LE.9) THEN

    FY=YY

ELSE

    FY=YY+1

END IF

FY=FY-75

END

SUBROUTINE WRITE\_LOSS(OUTFILE,OUT,FIRST)

STRUCTURE /LOSS\_FILE/

    INTEGER SSN,STR\_LOSS,LOSS\_REASON

    CHARACTER\*2 RQC

    INTEGER PROGRAM,RATING,PAYGRADE,LOS

END STRUCTURE

RECORD /LOSS\_FILE/ OUT

INTEGER OUTFILE,FIRST

IF (FIRST.EQ.1) THEN

    WRITE(OUTFILE,10)

    FORMAT(' ')

END IF

WRITE(OUTFILE,110) OUT.SSN,OUT.STR\_LOSS,OUT.LOSS\_REASON,OUT.RQC,

    OUT.PROGRAM,OUT.RATING,OUT.PAYGRADE,OUT.LOS

FORMAT(1X,I9,1X,I4,I3,A2,4I3)

RETURN

END

```

SUBROUTINE WRITE_LONG(OUTFILE,OUT,FIRST)

STRUCTURE /LONGIT_FILE/
  INTEGER SSN,YRFLAG(11),RATING(11),PAYGRADE(11)
END STRUCTURE
RECORD /LONGIT_FILE/ OUT

INTEGER OUTFILE,FIRST
INTEGER T

IF (FIRST.EQ.1) THEN
  WRITE(OUTFILE,10)
  FORMAT(' ')
END IF

WRITE(OUTFILE,100) OUT.SSN,(OUT.YRFLAG(T),T=1,11),
*              (OUT.RATING(T),OUT.PAYGRADE(T),T=1,11)
100 FORMAT(1X,I9,1X,11I1,11(1X,I4,I1))

RETURN
END

```



```

SUBROUTINE TYPE_GAIN(CURR, LONG, LOSS_YR_A, I, UP,
*                   T_TYPE, R_TYPE, KRATE, PG, LOSS_CNT)

IMPLICIT NONE

STRUCTURE /LOSS_FILE/
  INTEGER SSN, STR_LOSS, LOSS_REASON
  CHARACTER*2 RQC
  INTEGER PROGRAM, RATING, PAYGRADE, LOS
END STRUCTURE
RECORD /LOSS_FILE/ CURR

STRUCTURE /LONGIT_FILE/
  INTEGER SSN, YRFLAG(11), RATING(11), PAYGRADE(11)
END STRUCTURE
RECORD /LONGIT_FILE/ LONG

INTEGER LOSS_YR_A, UP, T_TYPE, R_TYPE, ST, KRATE, PG, LOSS_CNT
INTEGER*2 I
INTEGER IO

IF (LOSS_YR_A.EQ.I) THEN
  T_TYPE=1
ELSE
  IF (LOSS_YR_A.EQ.I-1) THEN
    T_TYPE=2
  ELSE
    T_TYPE=3
  END IF
END IF

PG=LONG.PAYGRADE(I)
ST=2
CALL KEVIN_TO_RATING(ST, LONG.RATING(I), KRATE, LOSS_CNT)
IF (ST.EQ.99) THEN
  DO IO=I, UP
    ST=2
    CALL KEVIN_TO_RATING(ST, LONG.RATING(IO), KRATE, LOSS_CNT)
    IF (ST.NE.99) THEN
      PG=LONG.PAYGRADE(IO)
      GOTO 50
    END IF
  END DO
  WRITE(6,*) 'TYPE_RATING>> NO RATING @', LOSS_CNT
  KRATE=96
END IF
50 CONTINUE

IF ((CURR.RATING.EQ. KRATE).AND.
*   (PG.EQ. CURR.PAYGRADE)) THEN
  R_TYPE=1
ELSE
  R_TYPE=2
END IF

RETURN
END

```

```

SUBROUTINE KEVIN_TO_RATING(STATE,RO,R,INCNT)

IMPLICIT NONE

INTEGER STATE,RO,R,INCNT

INTEGER KEVIN(140),RATING(140)
INTEGER I

CHARACTER*3 RATE_NAME(109)
COMMON /KEVIN_RATING/RATE_NAME

C INITIALIZE
  IF (STATE.EQ.0) GOTO 1000

C CONVERT KEVIN RATINGS TO 4-DIGIT RATING CODE
  IF (STATE.EQ.1) GOTO 2000

C CONVERT 4-DIGIT RATING CODES TO KEVIN RATING
  IF (STATE.EQ.2) GOTO 3000

C CONVERT OLD-KEVIN RATING (1..129) TO Total-Force RATING (1..108,109=OTH)
C                                     or TO Total-Force RATING (1..69,70=OTH)
  IF (STATE.EQ.3) GOTO 4000

  WRITE(6,100) STATE
100  FORMAT(1X,'CHECK RATING',BAD STATE PASSED; STATE=',I4)
  RETURN

C INITIALIZE
1000 CONTINUE
  I=0

1010 CONTINUE
  I=I+1
  READ(3,1100,END=1900) KEVIN(I),RATE_NAME(KEVIN(I)),RATING(I)
1100  FORMAT(X,I3,2X,A3,1X,I4)
  GOTO 1010

1900 CONTINUE
  CLOSE(1)
  RETURN

C CONVERT KEVIN RATINGS TO 4-DIGIT RATING CODE
2000 CONTINUE

  DO I=1,129
    IF (RO.EQ.KEVIN(I)) THEN
      R=RATING(I)
      GOTO 2100
    END IF
  END DO
  WRITE(6,*) 'CHECK RATING',BAD RATING: ',RO,' @ ',INCNT
2100 CONTINUE
  RETURN

```

# CONVERT 4-DIGIT RATING CODES TO KEVIN RATING

000 CONTINUE

DO I=1,129

IF (RO.EQ.RATING(I)) THEN

R=KEVIN(I)

GOTO 3100

END IF

END DO

IF (RO.EQ.9909) STATE=99

WRITE(6,\*) 'CHECK RATING' BAD RATING: ',RO,' @ ',INCNT

000 CONTINUE

RETURN

## CONVERT OLD-KEVIN RATING TO NEW KEVIN-RATING

000 CONTINUE

IF ((1.LE.RO).AND.(RO.LE.129)) THEN

R=KEVIN( RO )

ELSE

WRITE(6,\*) 'CHECK RATING' BAD OLD-KEVIN RATING ',RO,' @ ',INCNT

END IF

RETURN

END

SUBROUTINE WRITE\_LINE(LABEL,ONE,TWO,THREE,FOUR)

CHARACTER\*3 LABEL

INTEGER\*4 ONE,TWO,THREE,FOUR

INTEGER\*4 NUM(2),DEN(2)

INTEGER P

NUM(1)=ONE

NUM(2)=TWO

DEN(1)=THREE

DEN(2)=FOUR

IF (DEN(1).NE.0) THEN

IF (DEN(2).NE.0) THEN

WRITE(7,100) LABEL,

\* (NUM(I),DEN(I),REAL(NUM(I))/REAL(DEN(I)),I=1,2)

ELSE

WRITE(7,110) LABEL,

\* NUM(1),DEN(1),REAL(NUM(1))/REAL(DEN(1)),NUM(2)

END IF

ELSE

IF (DEN(2).NE.0) THEN

WRITE(7,120) LABEL,

\* NUM(1),NUM(2),DEN(2),REAL(NUM(2))/REAL(DEN(2))

ELSE

WRITE(7,130) LABEL,NUM(1),NUM(2)

END IF

END IF

```
100  FORMAT(1X,A3,2(3X,I5,3X,I5,3X,F4.2))
110  FORMAT(1X,A3,3X,I5,3X,I5,3X,F4.2,3X,I5,7X,'0',3X,'----','<----')
120  FORMAT(1X,A3,3X,I5,7X,'0',3X,'----',3X,I5,3X,I5,3X,F4.2,'<----')
130  FORMAT(1X,A3,3X,I5,7X,'0',3X,'----',3X,I5,7X,'0',3X,'----','<==')

      RETURN
      END
```

ANNEX C-2

PROGRAM LISTINGS FOR TABULATING SELRES INVENTORY

IDENTIFICATION DIVISION.  
PROGRAM-ID. INV84-AGG.  
AUTHOR. J GROGAN.  
INSTALLATION. CNA.  
DATE-WRITTEN. 11/06/86.  
DATE-COMPILED.  
SECURITY. UNCLASSIFIED.

ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER. VAX-11-780.  
OBJECT-COMPUTER. VAX-11-780.

INPUT-OUTPUT SECTION.  
FILE-CONTROL.

SELECT INFILE	ASSIGN TO IFILE.
SELECT INFILE2	ASSIGN TO IFILE2.
SELECT OUTFILE1	ASSIGN TO OFILE1.
SELECT OUTFILE2	ASSIGN TO OFILE2.
SELECT PRINTFILE	ASSIGN TO PFILE.

DATA DIVISION.  
FILE SECTION.

FD INFILE  
RECORD CONTAINS 14 CHARACTERS  
DATA RECORD IS INREC.

01 INREC.  
05 FILLER PIC X.  
05 RATE-NUM-IN PIC Z(3).  
05 FILLER PIC X(2).  
05 ALPHA-RATE-IN PIC X(3).  
05 FILLER PIC X(1).  
05 RATECODE-IN PIC 9(4).

FD INFILE2  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS INREC2.

01 INREC2.  
05 SSN PIC X(9).  
05 YEAR-FLAG OCCURS 11 TIMES  
PIC 9(1).  
05 FILLER PIC X(54).  
05 RCODE PIC Z(3)9.  
05 FILLER PIC X(8).

FD OUTFILE1  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS OUTREC1.

01 OUTREC1.  
05 FILLER PIC X(86).

FD OUTFILE2  
RECORD CONTAINS 51 CHARACTERS  
DATA RECORD IS OUTREC2.

01 OUTREC2.  
 05 RATING-OUT PIC X(3).  
 05 LOS-DATA OCCURS 8 TIMES.  
 10 FILLER PIC X(2).  
 10 INV-OUT PIC 9(4).

FD PRINTFILE  
 DATA RECORD IS PRINTLINE.

01 PRINTLINE PIC X(80).

# WORKING-STORAGE SECTION.

01 FILE-COUNTERS.  
 05 PARCNT PIC 9(4) USAGE COMP.  
 05 EOF PIC 9(1) USAGE COMP.  
 05 EOF2 PIC 9(1) USAGE COMP.  
 05 INCNT PIC 9(6) USAGE COMP.  
 05 INCNT2 PIC 9(6) USAGE COMP.  
 05 OUTCNT1 PIC 9(6) USAGE COMP.  
 05 OUTCNT2 PIC 9(6) USAGE COMP.

01 STATEMENT.  
 05 VAR-LABEL PIC X(35).  
 05 VAR-NUMBER-1 PIC Z(1)9(1).  
 05 VAR-NUMBER PIC Z(9)9(1).

01 JIM-COUNTERS.  
 05 TOTAL PIC 9(7) USAGE COMP.  
 05 NO-MATCH PIC 9(7) USAGE COMP.  
 05 PTR PIC 9(3).  
 05 RATE-NUM-IN-T PIC 9(3).  
 05 RCODE-T PIC 9(4).  
 05 YEAR PIC S9(2).  
 05 RNUM PIC 9(3).  
 05 GAIN-YEAR PIC 9(2).  
 05 LOS PIC 9(2).  
 05 DONE PIC 9(1).  
 05 D-CNT PIC 9(6) USAGE COMP.

01 RATECODE-TABLE.  
 05 RATECODE-DATA OCCURS 129 TIMES  
 ASCENDING KEY IS RATECODE  
 INDEXED BY R-INDX.  
 10 RATE-NUM PIC 9(3).  
 10 RATECODE PIC 9(4).  
 10 ALPHA-RATE PIC X(3).

01 NAVET-TABLE.  
 05 NAVET-RATE OCCURS 70 TIMES.  
 10 INV OCCURS 8 TIMES PIC 9(6).

PROCEDURE DIVISION.  
MAIN SECTION.

OVERALL-STRUCTURE.

PERFORM 100-BEGINNING.  
PERFORM 325-SETUP-RATE-TABLE.  
PERFORM 400-MAIN-FILE-LOGIC  
UNTIL (EOF2 = 1).  
PERFORM 1000-TERMINATION.  
STOP RUN.

EXIT-OVER-STRUCTURE.  
EXIT.

100-BEGINNING.

OPEN INPUT INFILE, INFILE2,  
OUTPUT OUTFILE1,OUTFILE2, PRINTFILE,  
INITIALIZE FILE-COUNTERS, JIM-COUNTERS, RATECODE-TABLE,NAVET-TABLE.  
MOVE SPACES TO OUTREC1,OUTREC2.  
PERFORM 300-READ-FILE2.

300-READ-FILE2.

READ INFILE2  
AT END MOVE 1 TO EOF2.  
IF (EOF2 NOT = 1)  
MOVE RCODE TO RCODE-T  
ADD 1 TO INCNT2.

325-SETUP-RATE-TABLE.

PERFORM 350-READ-INFILE.  
MOVE 1 TO PTR.  
PERFORM 375-READ-IN-DATA-FOR-TABLE  
UNTIL (EOF = 1).  
CLOSE INFILE.  
MOVE 70 TO RATE-NUM(129)  
MOVE 9909 TO RATECODE(129).  
MOVE 'OTH' TO ALPHA-RATE(70).

350-READ-INFILE.

READ INFILE  
AT END MOVE 1 TO EOF.

375-READ-IN-DATA-FOR-TABLE.

ADD 1 TO INCNT.  
MOVE RATECODE-IN TO RATECODE(PTR).  
MOVE RATE-NUM-IN TO RATE-NUM(PTR).  
MOVE RATE-NUM-IN TO RATE-NUM-IN-T  
IF (ALPHA-RATE(RATE-NUM-IN-T) = " ")  
MOVE ALPHA-RATE-IN TO ALPHA-RATE(RATE-NUM-IN-T).  
PERFORM 350-READ-INFILE.  
ADD 1 TO PTR.

400-MAIN-FILE-LOGIC.

IF (YEAR-FLAG(10)=1)  
PERFORM 500-GET-GAIN-YR  
ADD 1 TO TOTAL



```

SEARCH ALL RATECODE-DATA
  AT END  PERFORM
    ADD 1 TO INV(RATE-NUM(129),LOS)
    SET R-INDX TO 129
    END-PERFORM
  WHEN (RATECODE(R-INDX) = RCODE-T)
    ADD 1 TO INV(RATE-NUM(R-INDX),LOS)
  END-SEARCH
  IF (R-INDX=129) AND (D-CNT < 100)
    DISPLAY INREC2
    ADD 1 TO D-CNT
  END-IF
ELSE
  ADD 1 TO NO-MATCH.

```

```

PERFORM 300-READ-FILE2.

```

```

500-GET-GAIN-YR.
  MOVE 1 TO DONE
  MOVE 9 TO YEAR
  PERFORM UNTIL (DONE=0)
    IF (YEAR=0)
      MOVE 0 TO DONE
    ELSE
      IF (YEAR-FLAG(YEAR)=0)
        MOVE 0 TO DONE
      END-IF
    END-IF
    SUBTRACT 1 FROM YEAR
  END-PERFORM.
  IF (YEAR=-1)
    MOVE 75 TO GAIN-YEAR
    MOVE 10 TO LOS
  ELSE
    ADD 75 2 YEAR GIVING GAIN-YEAR
    SUBTRACT YEAR FROM 9 GIVING LOS.
  IF (LOS>8)
    MOVE 8 TO LOS.

```

```

700-WRITE-LONGIT.
  MOVE INREC2 TO OUTREC1.
  WRITE OUTREC1.
  ADD 1 TO OUTCNT1.

```

```

1000-TERMINATION.
  PERFORM 1200-WRITE-TABLE.
  PERFORM 1100-WRITE-RESULTS.
  CLOSE INFILE2, OUTFILE1,OUTFILE2, PRINTFILE.

```

1100-WRITE-RESULTS.

MOVE "NUMBER OF OTHERS IN '85 INVENTORY" TO VAR-LABEL.  
MOVE TOTAL TO VAR-NUMBER.  
WRITE PRINTLINE FROM STATEMENT AFTER 2 LINES.  
MOVE "NUMBER OF NON-85 OTHERS" TO VAR-LABEL.  
MOVE NO-MATCH TO VAR-NUMBER.  
WRITE PRINTLINE FROM STATEMENT.  
MOVE "NUMBER OF INFILE2 RECORDS" TO VAR-LABEL.  
MOVE INCNT2 TO VAR-NUMBER.  
WRITE PRINTLINE FROM STATEMENT .  
MOVE "NUMBER OF OUTFILE1 RECORDS " TO VAR-LABEL.  
MOVE OUTCNT1 TO VAR-NUMBER.  
WRITE PRINTLINE FROM STATEMENT.

1200-WRITE-TABLE.

MOVE "APG/OSVET INVENTORY" TO VAR-LABEL.  
MOVE 0 TO VAR-NUMBER.  
WRITE OUTREC2 FROM STATEMENT AFTER 2 LINES.  
MOVE SPACES TO OUTREC2.  
PERFORM VARYING RNUM FROM 1 BY 1 UNTIL (RNUM>70)  
MOVE ALPHA-RATE(RNUM) TO RATING-OUT  
PERFORM VARYING LOS FROM 1 BY 1 UNTIL (LOS>8)  
MOVE INV(RNUM,LOS) TO INV-OUT(LOS)  
END-PERFORM  
WRITE OUTREC2  
END-PERFORM.

IDENTIFICATION DIVISION.  
PROGRAM-ID. INV84-AGG.  
AUTHOR. J GROGAN.  
INSTALLATION. CNA.  
DATE-WRITTEN. 11/06/86.  
DATE-COMPILED.  
SECURITY. UNCLASSIFIED.

ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER. VAX-11-780.  
OBJECT-COMPUTER. VAX-11-780.

INPUT-OUTPUT SECTION.  
FILE-CONTROL.

SELECT INFILE	ASSIGN TO IFILE.
SELECT INFILE1	ASSIGN TO IFILE1.
SELECT INFILE2	ASSIGN TO IFILE2.
SELECT OUTFILE1	ASSIGN TO OFILE1.
SELECT OUTFILE2	ASSIGN TO OFILE2.
SELECT PRINTFILE	ASSIGN TO PFILE.

DATA DIVISION.  
FILE SECTION.

FD INFILE  
RECORD CONTAINS 14 CHARACTERS  
DATA RECORD IS INREC.

01 INREC.  
05 FILLER PIC X.  
05 RATE-NUM-IN PIC Z(3).  
05 FILLER PIC X(2).  
05 ALPHA-RATE-IN PIC X(3).  
05 FILLER PIC X(1).  
05 RATECODE-IN PIC 9(4).

FD INFILE1  
RECORD CONTAINS 22 CHARACTERS  
DATA RECORD IS INREC1.

01 INREC1.  
05 SAM-SSN PIC 9(9).  
05 FILLER PIC X(7).  
05 GAIN-DATE PIC 9(4).  
05 FILLER PIC X(2).

FD INFILE2  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS INREC2.

01 INREC2.  
05 SSN PIC X(9).  
05 YEAR-FLAG OCCURS 11 TIMES  
PIC 9(1).  
05 FILLER PIC X(54).  
05 RCODE PIC Z(3)9.  
05 FILLER PIC X(8).

FD OUTFILE1  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS OUTREC1.

01 OUTREC1.  
05 FILLER PIC X(86).

FD OUTFILE2  
RECORD CONTAINS 51 CHARACTERS  
DATA RECORD IS OUTREC2.

01 OUTREC2.  
05 RATING-OUT PIC X(3).  
05 LOS-DATA OCCURS 8 TIMES.  
10 FILLER PIC X(2).  
10 INV-OUT PIC 9(4).

FD PRINTFILE  
DATA RECORD IS PRINTLINE.

01 PRINTLINE PIC X(80).

#### WORKING-STORAGE SECTION.

01 FILE-COUNTERS.  
05 PARCNT PIC 9(4) USAGE COMP.  
05 EOF PIC 9(1) USAGE COMP.  
05 EOF1 PIC 9(1) USAGE COMP.  
05 EOF2 PIC 9(1) USAGE COMP.  
05 INCNT PIC 9(6) USAGE COMP.  
05 INCNT1 PIC 9(6) USAGE COMP.  
05 INCNT2 PIC 9(6) USAGE COMP.  
05 OUTCNT1 PIC 9(6) USAGE COMP.  
05 OUTCNT2 PIC 9(6) USAGE COMP.

01 STATEMENT.  
05 VAR-LABEL PIC X(35).  
05 VAR-NUMBER-1 PIC Z(1)9(1).  
05 VAR-NUMBER PIC Z(9)9(1).

01 JIM-COUNTERS.  
05 TEMP PIC 9(6).  
05 TOTAL PIC 9(7) USAGE COMP.  
05 MATCHED PIC 9(7) USAGE COMP.  
05 MATCHES-85 PIC 9(7) USAGE COMP.  
05 PTR PIC 9(3).  
05 RATE-NUM-IN-T PIC 9(3).  
05 RCODE-T PIC 9(4).  
05 YEAR PIC S9(2).  
05 YEAR-1 PIC 9(2).  
05 CHG-CNT PIC 9(6) USAGE COMP.  
05 RNUM PIC 9(3).  
05 PROG PIC 9(1).  
05 GAIN-YEAR PIC 9(2).  
05 LOS PIC 9(2).  
05 DONE PIC 9(1).  
05 D-CNT PIC 9(6) USAGE COMP.  
05 TOTAL-84 PIC 9(6) USAGE COMP.

```

01 RATECODE-TABLE.
   05 RATECODE-DATA OCCURS 129 TIMES
      ASCENDING KEY IS RATECODE
      INDEXED BY R-INDX.
      10 RATE-NUM          PIC 9(3).
      10 RATECODE          PIC 9(4).
      10 ALPHA-RATE        PIC X(3).

01 NAVET-TABLE.
   05 NAVET-RATE OCCURS 70 TIMES.
      10 INV      OCCURS 8 TIMES  PIC 9(4).

```

```

PROCEDURE DIVISION.
DECLARATIVES.
IOERROR SECTION.
    USE AFTER ERROR PROCEDURE ON INFILE.
PARITY-ACTION.
    ADD 1 TO PARCNT.
    IF (PARCNT > 2000)
        DISPLAY " ABORTING FROM DECLARATIVES SECTION. "
        STOP RUN.
END DECLARATIVES.

```

MAIN SECTION.

```

OVERALL-STRUCTURE.
    PERFORM 100-BEGINNING.
    PERFORM 325-SETUP-RATE-TABLE.
    PERFORM 400-MAIN-FILE-LOGIC
        UNTIL (EOF1 = 1) OR (EOF2 = 1).
    PERFORM 1000-TERMINATION.
    STOP RUN.
EXIT-OVER-STRUCTURE.
    EXIT.

```

```

100-BEGINNING.
    OPEN INPUT  INFILE, INFILE1,INFILE2,
        OUTPUT  OUTFILE1,OUTFILE2, PRINTFILE,
        INITIALIZE FILE-COUNTERS, JIM-COUNTERS, RATECODE-TABLE,NAVET-TABLE.
    MOVE SPACES TO OUTREC1,OUTREC2.
    PERFORM 200-READ-FILE1.
    PERFORM 300-READ-FILE2.

```

```

200-READ-FILE1.
    READ INFILE1
        AT END MOVE 1 TO EOF1.
    IF (EOF1 NOT = 1)
        ADD 1 TO INCNT1..

```

```

300-READ-FILE2.
    READ INFILE2
        AT END MOVE 1 TO EOF2.
    IF (EOF2 NOT = 1)
        MOVE RCODE TO RCODE-T
        ADD 1 TO INCNT2.

```

# 325-SETUP-RATE-TABLE.

```

PERFORM 350-READ-INFILE.
MOVE 1 TO PTR.
PERFORM 375-READ-IN-DATA-FOR-TABLE
    UNTIL (EOF = 1).
CLOSE INFILE.
MOVE 70 TO RATE-NUM(129)
MOVE 9909 TO RATECODE(129).
MOVE 'OTH' TO ALPHA-RATE(70).

```

# 350-READ-INFILE.

```

READ INFILE
    AT END MOVE 1 TO EOF.

```

# 375-READ-IN-DATA-FOR-TABLE.

```

ADD 1 TO INCNT.
MOVE RATECODE-IN TO RATECODE(PTR).
MOVE RATE-NUM-IN TO RATE-NUM(PTR).
MOVE RATE-NUM-IN TO RATE-NUM-IN-T
IF (ALPHA-RATE(RATE-NUM-IN-T) = " ")
    MOVE ALPHA-RATE-IN TO ALPHA-RATE(RATE-NUM-IN-T).
PERFORM 350-READ-INFILE.
ADD 1 TO PTR.

```

# 400-MAIN-FILE-LOGIC.

```

IF ( SAM-SSN < SSN )
    PERFORM 200-READ-FILE1
ELSE
    IF (SAM-SSN = SSN )
        ADD 1 TO MATCHED
        IF (YEAR-FLAG(9)=1)
            ADD 1 TO TOTAL-84
        END-IF
        IF (YEAR-FLAG(10)=1)
            PERFORM 500-GET-GAIN-YR
            ADD 1 TO TOTAL
            SEARCH ALL RATECODE-DATA
                AT END PERFORM
                    ADD 1 TO INV(RATE-NUM(129),LOS)
                    SET R-INDX TO 129
                    END-PERFORM
                WHEN (RATECODE(R-INDX) = RCODE-T)
                    ADD 1 TO INV(RATE-NUM(R-INDX),LOS)
            END-SEARCH
            IF (R-INDX=129) AND (D-CNT < 100)
                DISPLAY INREC2
                ADD 1 TO D-CNT
            END-IF
        ELSE
            PERFORM 700-WRITE-LONGIT
        END-IF
    PERFORM 200-READ-FILE1
    PERFORM 300-READ-FILE2
ELSE
    *
    NAV-SSN > SSN
    PERFORM 700-WRITE-LONGIT
    PERFORM 300-READ-FILE2.

```

```

IF (EOF1=1)
  PERFORM UNTIL (EOF2=1)
    PERFORM 700-WRITE-LONGIT
    PERFORM 300-READ-FILE2
  END-PERFORM.

```

500-GET-GAIN-YR.

\* cheating, we are only using FY'84.FY'85 data...

```

  MOVE 1 TO DONE
  MOVE 9 TO YEAR
  PERFORM UNTIL (DONE=0)
    IF (YEAR=0)
      MOVE 0 TO DONE
    ELSE
      IF (YEAR-FLAG(YEAR)=0)
        MOVE 0 TO DONE
      END-IF
    END-IF
  SUBTRACT 1 FROM YEAR
END-PERFORM.
IF (YEAR=-1)
  MOVE 75 TO GAIN-YEAR
  MOVE 10 TO LOS
ELSE
  ADD 75 2 YEAR GIVING GAIN-YEAR
  SUBTRACT YEAR FROM 9 GIVING LOS.
IF (LOS>8)
  MOVE 8 TO LOS.

```

700-WRITE-LONGIT.

```

  MOVE INREC2 TO OUTREC1.
  WRITE OUTREC1.
  ADD 1 TO OUTCNT1.

```

1000-TERMINATION.

```

  PERFORM 1200-WRITE-TABLE.
  PERFORM 1100-WRITE-RESULTS.
  CLOSE INFILE1, INFILE2, OUTFILE1,OUTFILE2, PRINTFILE.

```

1100-WRITE-RESULTS.

```

  MOVE "MATCHES WITH '84 SAMS" TO VAR-LABEL
  MOVE TOTAL-84 TO VAR-NUMBER
    WRITE PRINTLINE FROM STATEMENT AFTER 2 LINES.
  MOVE "NUMBER OF SAMS IN INVENTORY" TO VAR-LABEL.
  MOVE TOTAL TO VAR-NUMBER.
    WRITE PRINTLINE FROM STATEMENT .
  MOVE "TOTAL NUMBER OF SAM MATCHES" TO VAR-LABEL.
  MOVE MATCHED TO VAR-NUMBER.
    WRITE PRINTLINE FROM STATEMENT.
  MOVE "NUMBER OF INFILE1 RECORDS" TO VAR-LABEL.
  MOVE INCNT1 TO VAR-NUMBER.
    WRITE PRINTLINE FROM STATEMENT .
  MOVE "NUMBER OF INFILE2 RECORDS" TO VAR-LABEL.
  MOVE INCNT2 TO VAR-NUMBER.
    WRITE PRINTLINE FROM STATEMENT .
  MOVE "NUMBER OF OUTFILE1 RECORDS " TO VAR-LABEL.
  MOVE OUTCNT1 TO VAR-NUMBER.
    WRITE PRINTLINE FROM STATEMENT.

```

```

1200-WRITE-TABLE.
  MOVE "SAM INVENTORY" TO VAR-LABEL.
  MOVE TEMP TO VAR-NUMBER.
  WRITE OUTREC2 FROM STATEMENT AFTER 2 LINES.
  MOVE SPACES TO OUTREC2.
  PERFORM VARYING RNUM FROM 1 BY 1 UNTIL (RNUM>70)
    MOVE ALPHA-RATE(RNUM) TO RATING-OUT
    PERFORM VARYING LOS FROM 1 BY 1 UNTIL (LOS>8)
      MOVE INV(RNUM,LOS) TO INV-OUT(LOS)
    END-PERFORM
  WRITE OUTREC2
END-PERFORM.

```



IDENTIFICATION DIVISION.  
PROGRAM-ID. INV84-AGG.  
AUTHOR. J GROGAN.  
INSTALLATION. CNA.  
DATE-WRITTEN. 11/06/86.  
DATE-COMPILED.  
SECURITY. UNCLASSIFIED.

ENVIRONMENT DIVISION.  
CONFIGURATION SECTION.  
SOURCE-COMPUTER. VAX-11-780.  
OBJECT-COMPUTER. VAX-11-780.

INPUT-OUTPUT SECTION.

FILE-CONTROL.  
SELECT INFILE ASSIGN TO IFILE.  
SELECT INFILE1 ASSIGN TO IFILE1.  
SELECT INFILE2 ASSIGN TO IFILE2.  
SELECT OUTFILE1 ASSIGN TO OFILE1.  
SELECT OUTFILE2 ASSIGN TO OFILE2.  
SELECT PRINTFILE ASSIGN TO PFILE.

DATA DIVISION.  
FILE SECTION.

FD INFILE  
RECORD CONTAINS 14 CHARACTERS  
DATA RECORD IS INREC.

01 INREC.  
05 FILLER PIC X.  
05 RATE-NUM-IN PIC Z(3).  
05 FILLER PIC X(2).  
05 ALPHA-RATE-IN PIC X(3).  
05 FILLER PIC X(1).  
05 RATECODE-IN PIC 9(4).

FD INFILE1  
RECORD CONTAINS 30 CHARACTERS  
DATA RECORD IS INREC1.

01 INREC1.  
05 NAV-SSN PIC 9(9).  
05 LOSS-DATE PIC 9(4).  
05 FILLER PIC X(5).  
05 E-PROGRAM PIC 9(3).  
05 FILLER PIC X(9).

FD INFILE2  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS INREC2.

01 INREC2.  
05 SSN PIC X(9).  
05 YEAR-FLAG OCCURS 11 TIMES  
PIC 9(1).  
05 FILLER PIC X(54).  
05 RCODE PIC Z(3)9.  
05 FILLER PIC X(8).

FD OUTFILE1  
RECORD CONTAINS 86 CHARACTERS  
DATA RECORD IS OUTREC1.

01 OUTREC1.  
05 FILLER PIC X(86).

FD OUTFILE2  
RECORD CONTAINS 51 CHARACTERS  
DATA RECORD IS OUTREC2.

01 OUTREC2.  
05 RATING-OUT PIC X(3).  
05 LOS-DATA OCCURS 8 TIMES.  
10 FILLER PIC X(2).  
10 INV-OUT PIC 9(4).

FD PRINTFILE  
DATA RECORD IS PRINTLINE.

01 PRINTLINE PIC X(80).

WORKING-STORAGE SECTION.

01 FILE-COUNTERS.  
05 PARCNT PIC 9(4) USAGE COMP.  
05 EOF PIC 9(1) USAGE COMP.  
05 EOF1 PIC 9(1) USAGE COMP.  
05 EOF2 PIC 9(1) USAGE COMP.  
05 INCNT PIC 9(6) USAGE COMP.  
05 INCNT1 PIC 9(6) USAGE COMP.  
05 INCNT2 PIC 9(6) USAGE COMP.  
05 OUTCNT1 PIC 9(6) USAGE COMP.  
05 OUTCNT2 PIC 9(6) USAGE COMP.

01 STATEMENT.  
05 VAR-LABEL PIC X(35).  
05 VAR-NUMBER-1 PIC Z(1)9(1).  
05 VAR-NUMBER PIC Z(9)9(1).

01 JIM-COUNTERS.  
05 TEMP PIC 9(6).  
05 TOTAL PIC 9(7) USAGE COMP.  
05 MATCHED PIC 9(7) USAGE COMP.  
05 MATCHES-85 PIC 9(7) USAGE COMP.  
05 PTR PIC 9(3).  
05 RATE-NUM-IN-T PIC 9(3).  
05 RCODE-T PIC 9(4).  
05 YEAR PIC S9(2).  
05 YEAR-1 PIC 9(2).  
05 CHG-CNT PIC 9(6) USAGE COMP.  
05 RNUM PIC 9(3).  
05 PROG PIC 9(1).  
05 GAIN-YEAR PIC 9(2).  
05 LOS PIC 9(2).  
05 DONE PIC 9(1).  
05 NAVET PIC 9(1).  
05 D-CNT PIC 9(6) USAGE COMP.  
05 PROG-2-CNT PIC 9(6) USAGE COMP.

```

01  OTHER-COUNTERS.
    05  UPPER.
        10  UPPER-YR    PIC 9(2).
        10  UPPER-MO    PIC 9(2) VALUE IS 9.
    05  LOWER.
        10  LOWER-YR    PIC 9(2).
        10  LOWER-MO    PIC 9(2) VALUE IS 10.

01  RATECODE-TABLE.
    05  RATECODE-DATA OCCURS 129 TIMES
        ASCENDING KEY IS RATECODE
        INDEXED BY R-INDX.
        10  RATE-NUM      PIC 9(3).
        10  RATECODE      PIC 9(4).
        10  ALPHA-RATE    PIC X(3).

01  NAVET-TABLE.
    05  NAVET-RATE OCCURS 70 TIMES.
        10  PROGRAM-DATA OCCURS 2 TIMES.
            15 INV    OCCURS 8 TIMES    PIC 9(4).

```

PROCEDURE DIVISION.

DECLARATIVES.

IOERROR SECTION.

USE AFTER ERROR PROCEDURE ON INFILE.

PARITY-ACTION.

ADD 1 TO PARCNT.

IF (PARCNT > 2000)

DISPLAY " ABORTING FROM DECLARATIVES SECTION. "

STOP RUN.

END DECLARATIVES.

MAIN SECTION.

OVERALL-STRUCTURE.

PERFORM 100-BEGINNING.

PERFORM 325-SETUP-RATE-TABLE.

PERFORM 400-MAIN-FILE-LOGIC

UNTIL (EOF1 = 1) OR (EOF2 = 1).

PERFORM 1000-TERMINATION.

STOP RUN.

EXIT-OVER-STRUCTURE.

EXIT.

100-BEGINNING.

OPEN INPUT INFILE, INFILE1, INFILE2,

OUTPUT OUTFILE1, OUTFILE2, PRINTFILE,

INITIALIZE FILE-COUNTERS, JIM-COUNTERS, RATECODE-TABLE, NAVET-TABLE.

MOVE SPACES TO OUTREC1, OUTREC2.

PERFORM 200-READ-FILE1.

PERFORM 300-READ-FILE2.

200-READ-FILE1.

PERFORM 210-READ-FILE1.

PERFORM 210-READ-FILE1 UNTIL (LOSS-DATE > 7809) OR (EOF1=1).

```

210-READ-FILE1.
  READ INFILE1
    AT END MOVE 1 TO EOF1.
  IF (EOF1 NOT = 1)
    ADD 1 TO INCNT1.

300-READ-FILE2.
  READ INFILE2
    AT END MOVE 1 TO EOF2.
  IF (EOF2 NOT = 1)
    MOVE RCODE TO RCODE-T
    ADD 1 TO INCNT2.

325-SETUP-RATE-TABLE.
  PERFORM 350-READ-INFILE.
  MOVE 1 TO PTR.
  PERFORM 375-READ-IN-DATA-FOR-TABLE
    UNTIL (EOF = 1).
  CLOSE INFILE.
  MOVE 70 TO RATE-NUM(129)
  MOVE 9909 TO RATECODE(129).
  MOVE 'OTH' TO ALPHA-RATE(70).

350-READ-INFILE.
  READ INFILE
    AT END MOVE 1 TO EOF.

375-READ-IN-DATA-FOR-TABLE.
  ADD 1 TO INCNT.
  MOVE RATECODE-IN TO RATECODE(PTR).
  MOVE RATE-NUM-IN TO RATE-NUM(PTR).
  MOVE RATE-NUM-IN TO RATE-NUM-IN-T
  IF (ALPHA-RATE(RATE-NUM-IN-T) = " ")
    MOVE ALPHA-RATE-IN TO ALPHA-RATE(RATE-NUM-IN-T).
  PERFORM 350-READ-INFILE.
  ADD 1 TO PTR.

400-MAIN-FILE-LOGIC.
  IF ( NAV-SSN < SSN )
    PERFORM 200-READ-FILE1
  ELSE
    IF (NAV-SSN = SSN )
      ADD 1 TO MATCHED
      IF (YEAR-FLAG(10)=1)
        ADD 1 TO MATCHES-85
        PERFORM 500-GET-GAIN-YR
        PERFORM 600-COMPARE-DATE
        IF (NAVET=1)
          IF (E-PROGRAM=2)
            MOVE 2 TO PROG
            ADD 1 TO PROG-2-CNT
          ELSE
            MOVE 1 TO PROG
        END-IF
      ADD 1 TO TOTAL

```

```

        SEARCH ALL RATECODE-DATA
          AT END  PERFORM
            ADD 1 TO INV(RATE-NUM(129),PROG,LOS)
            SET R-INDX TO 129
            END-PERFORM
          WHEN (RATECODE(R-INDX) = RCODE-T)
            ADD 1 TO INV(RATE-NUM(R-INDX),PROG,LOS)
          END-SEARCH
        IF (R-INDX=129) AND (D-CNT < 100)
          DISPLAY INREC2
          ADD 1 TO D-CNT
        END-IF
      ELSE
        PERFORM 700-WRITE-LONGIT
      END-IF
    ELSE
      PERFORM 700-WRITE-LONGIT
    END-IF
    PERFORM 200-READ-FILE1
    PERFORM 300-READ-FILE2
  ELSE
    * NAV-SSN > SSN
    PERFORM 700-WRITE-LONGIT
    PERFORM 300-READ-FILE2.

  IF (EOF1=1)
    PERFORM UNTIL (EOF2=1)
      PERFORM 700-WRITE-LONGIT
      PERFORM 300-READ-FILE2
    END-PERFORM.

500-GET-GAIN-YR.
  MOVE 1 TO DONE
  MOVE 9 TO YEAR
  PERFORM UNTIL (DONE=0)
    IF (YEAR=0)
      MOVE 0 TO DONE
    ELSE
      IF (YEAR-FLAG(YEAR)=0)
        MOVE 0 TO DONE
      END-IF
    END-IF
    SUBTRACT 1 FROM YEAR
  END-PERFORM.
  IF (YEAR=-1)
    MOVE 75 TO GAIN-YEAR
    MOVE 10 TO LOS
  ELSE
    ADD 75 2 YEAR GIVING GAIN-YEAR
    SUBTRACT YEAR FROM 9 GIVING LOS.
  IF (LOS>8)
    MOVE 8 TO LOS.

```

```

600-COMPARE-DATE.
    MOVE GAIN-YEAR TO UPPER-YR.
    SUBTRACT 2 FROM UPPER-YR GIVING LOWER-YR.
    IF (LOWER-YR<78)
        MOVE 78 TO LOWER-YR.
    IF (LOWER<LOSS-DATE) AND (LOSS-DATE<UPPER)
        MOVE 1 TO NAVET
    ELSE
        MOVE 0 TO NAVET.

700-WRITE-LONGIT.
    MOVE INREC2 TO OUTREC1.
    WRITE OUTREC1.
    ADD 1 TO OUTCNT1.

1000-TERMINATION.
    PERFORM 1200-WRITE-TABLE.
    PERFORM 1100-WRITE-RESULTS.
    CLOSE INFILE1, INFILE2, OUTFILE1,OUTFILE2, PRINTFILE.

1100-WRITE-RESULTS.
    MOVE "NUMBER OF ACTIVE MARINERS" TO VAR-LABEL.
    MOVE PROG-2-CNT TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT.
    MOVE "NUMBER OF '85 RECENT NAVETS" TO VAR-LABEL.
    MOVE TOTAL TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT AFTER 2 LINES.
    MOVE "NUMBER OF '85 NAVET MATCHES" TO VAR-LABEL.
    MOVE MATCHES-85 TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT.
    MOVE "TOTAL NUMBER OF NAVET MATCHES" TO VAR-LABEL.
    MOVE MATCHED TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT.
    MOVE "NUMBER OF INFILE1 RECORDS" TO VAR-LABEL.
    MOVE INCNT1 TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT .
    MOVE "NUMBER OF INFILE2 RECORDS" TO VAR-LABEL.
    MOVE INCNT2 TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT .
    MOVE "NUMBER OF OUTFILE1 RECORDS " TO VAR-LABEL.
    MOVE OUTCNT1 TO VAR-NUMBER.
        WRITE PRINTLINE FROM STATEMENT.

1200-WRITE-TABLE.
    MOVE "4YO,5/6YO,PS,TAR INVENTORY" TO VAR-LABEL.
    MOVE TEMP TO VAR-NUMBER.
        WRITE OUTREC2 FROM STATEMENT AFTER 2 LINES.
    MOVE SPACES TO OUTREC2.
    PERFORM VARYING RNUM FROM 1 BY 1 UNTIL (RNUM<70)
        MOVE ALPHA-RATE(RNUM) TO RATING-OUT
        PERFORM VARYING LOS FROM 1 BY 1 UNTIL (LOS<8)
            MOVE INV(RNUM,1,LOS) TO INV-OUT(LOS)
        END-PERFORM
    WRITE OUTREC2
END-PERFORM.
    MOVE "ACTIVE MARINER INVENTORY" TO VAR-LABEL.
    MOVE TEMP TO VAR-NUMBER.
        WRITE OUTREC2 FROM STATEMENT AFTER 2 LINES.

```

```
MOVE SPACES TO OUTREC2.  
PERFORM VARYING RNUM FROM 1 BY 1 UNTIL (RNUM>70)  
    MOVE ALPHA-RATE(RNUM) TO RATING-OUT  
    PERFORM VARYING LOS FROM 1 BY 1 UNTIL (LOS>8)  
        MOVE INV(RNUM,2,LOS) TO INV-OUT(LOS)  
    END-PERFORM  
    WRITE OUTREC2  
END-PERFORM.
```

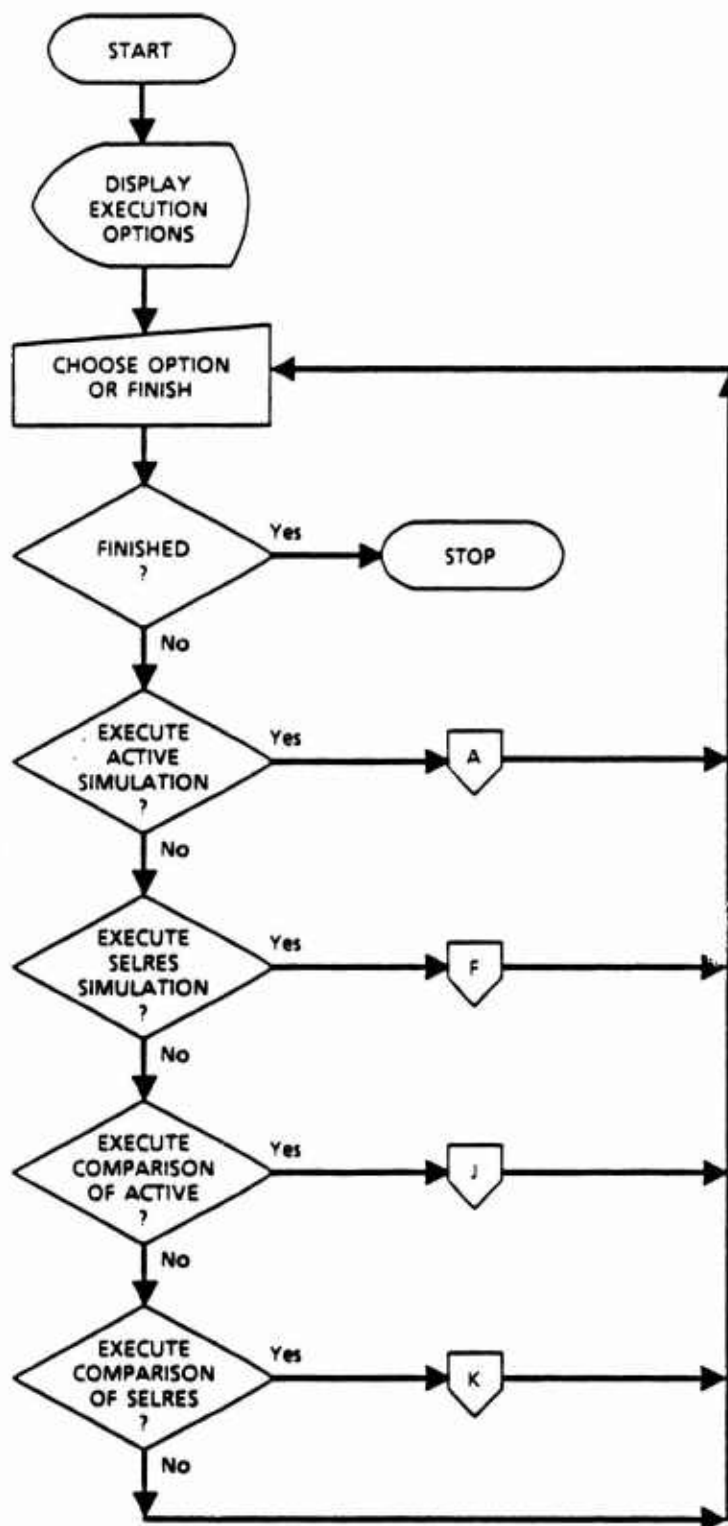
**APPENDIX D**  
**SIMULATION PROGRAMS**



## APPENDIX D

### SIMULATION PROGRAMS

This appendix lists the annotated simulation programs that simulate future active and reserve enlisted inventories. The flow chart in figure D-1 presents the logic of the simulation.



**FIG. D-1: SIMULATION FLOWCHART OF TOTAL FORCE ENLISTMENT PROGRAMS**

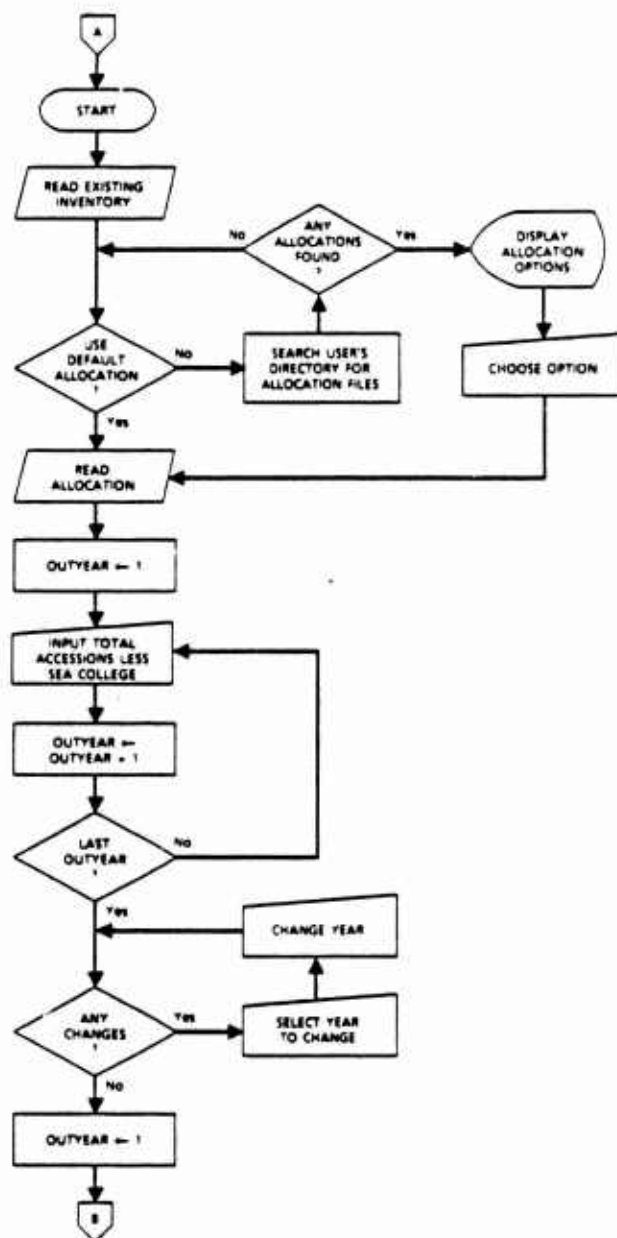


FIG. D-1: (Continued)

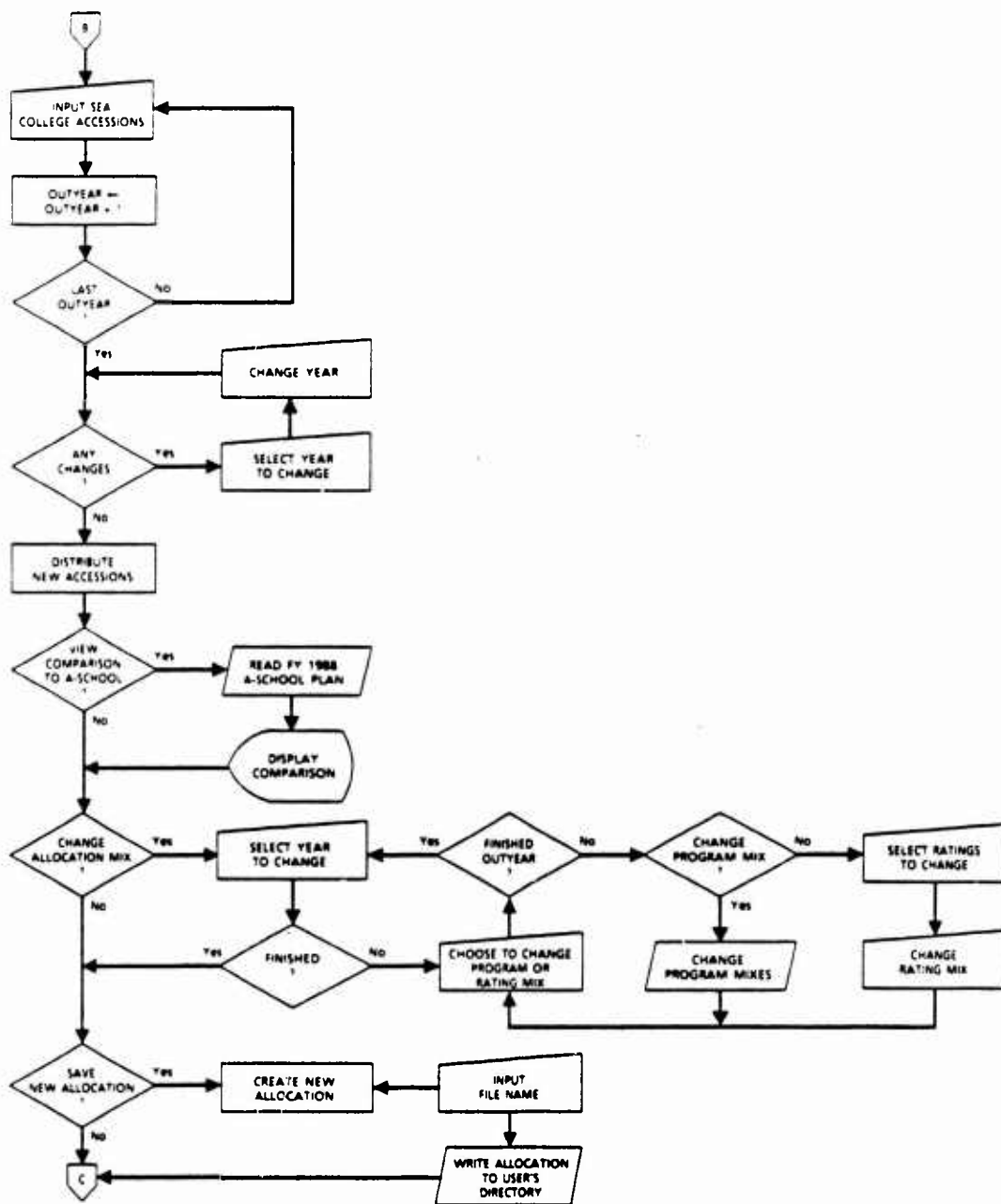


FIG. D-1: (Continued)

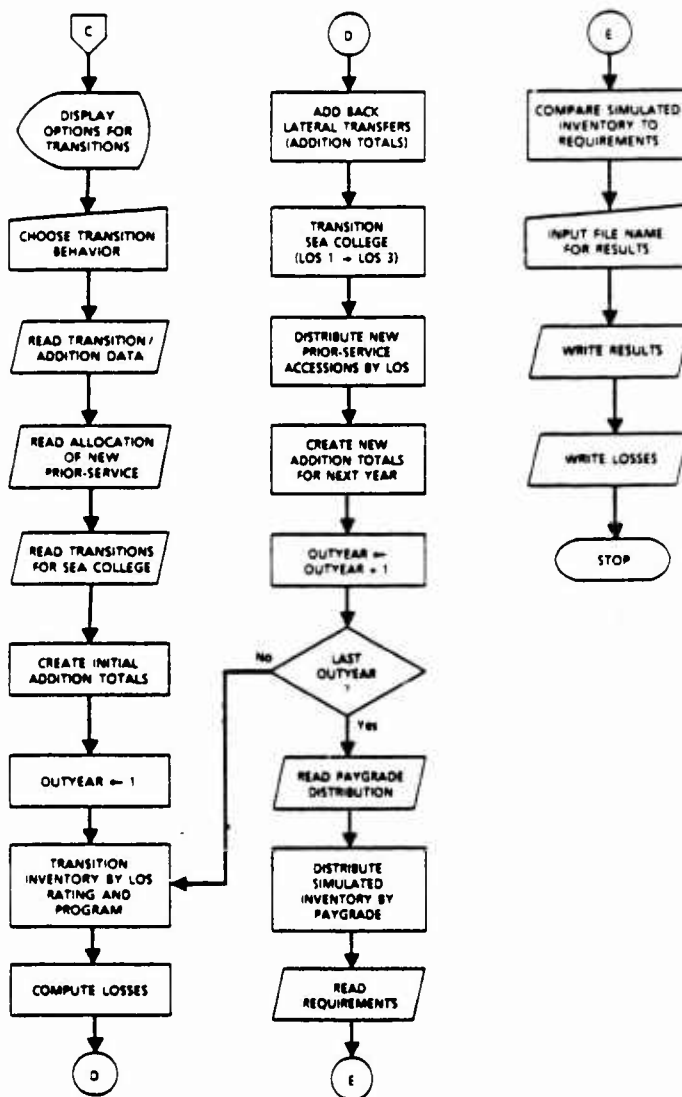


FIG. D-1: (Continued)

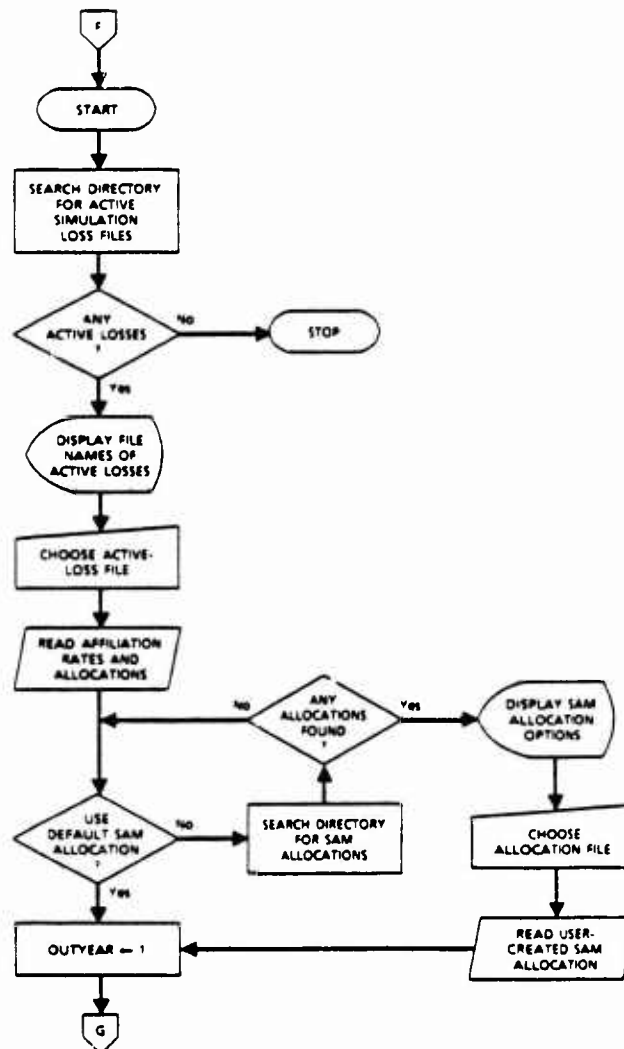


FIG. D-1: (Continued)

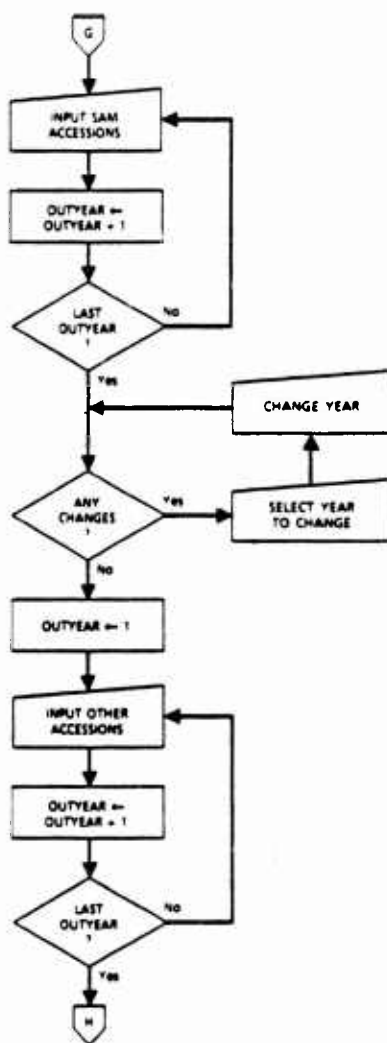


FIG. D-1: (Continued)

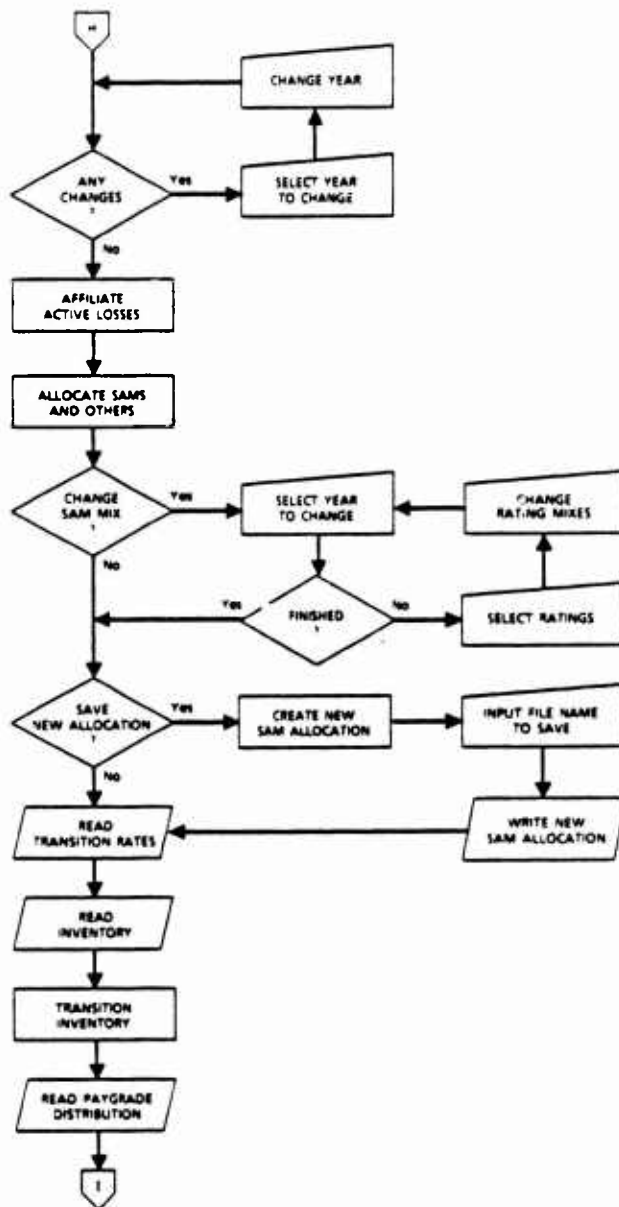


FIG. D-1: (Continued)



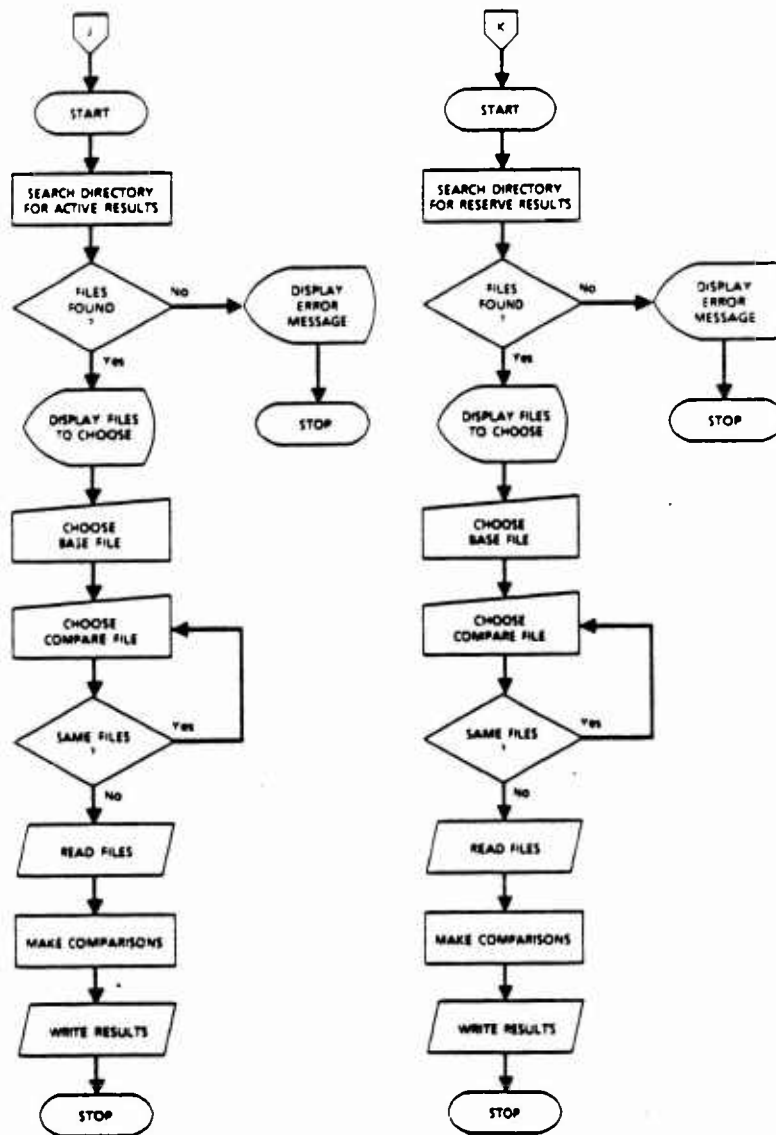


FIG. D-1: (Continued)

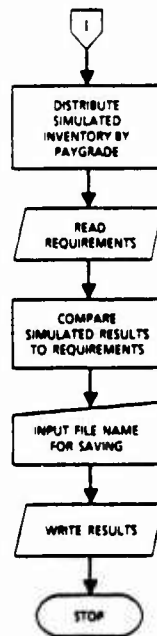


FIG. D-1: (Continued)

ANNEX D-1

LISTING OF COMMAND PROCEDURE TO EXECUTE SIMULATION PROGRAMS

```

$ ASSIGN/USERMODE SYS$COMMAND SYS$INPUT
$ DEFINE VT100 SYS$SYSDEVICE:[VT100]
$ @VT100:VT100
$!
$!
$ ON CONTROL_Y THEN GOTO PLAY_IT_AGAIN
$! 1          2          3          4          5          6          7
$!123456789 123456789 123456789 123456789 123456789 123456789 123456789 12345678
$!
$ PLAY_IT_AGAIN:
$!
$ NORMAL
$ CLEAR
$!
$ CURSOR 6 10 "1 - Execute Active simulation program"
$ CURSOR 8 10 "2 - Execute Reserve simulation program"
$ CURSOR 10 10 "3 - Execute program to compare two Active runs"
$ CURSOR 12 10 "4 - Execute program to compare two Reserve runs"
$ CURSOR 14 10 "9 - Finish this session"
$!
$!
$!
$ MESSED_UP:
$!
$ CURSOR 16 1 " "
$ INQUIRE XX "          Enter your selection"
$!
$!
$ IF XX .EQ. 9 THEN GOTO ENDIT
$ IF XX .EQ. 1 THEN GOTO RUN_ACTIVE
$ IF XX .EQ. 2 THEN GOTO RUN_RESERVE
$ IF XX .EQ. 3 THEN GOTO RUN_COMPARE_ACTIVE
$ IF XX .EQ. 4 THEN GOTO RUN_COMPARE_RESERVE
$!
$!
$ GOTO MESSED_UP
$!
$!
$!
$!
$!
$ RUN_ACTIVE:
$!          RUN CNA2:[CORLISSG.FORCE]ACTIVE
$          RUN CNA2:[CORLISSG.FORCE.NEW]ACTIVE
$ GOTO PLAY_IT_AGAIN
$!
$!
$!
$ RUN_RESERVE:
$          RUN CNA2:[CORLISSG.FORCE]RESERVE
$ GOTO PLAY_IT_AGAIN
$!
$!
$!
$ RUN_COMPARE_ACTIVE:
$          RUN CNA2:[CORLISSG.FORCE]COMPARE_A
$ GOTO PLAY_IT_AGAIN
$!

```

```
$!  
$!  
$ RUN_COMPARE_RESERVE:  
$      RUN CNA2:[CORLISSG.FORCE]COMPARE_R  
$ GOTO PLAY_IT_AGAIN  
$!  
$!  
$!  
$!  
$ ENDIT:  
$!  
$ CLEAR  
$!  
$ STOP
```

ANNEX D-2

LISTING OF ACTIVE-DUTY SIMULATION PROGRAM  
(VAX-11 FORTRAN)

---

E N L I S T M E N T   P R O G R A M S   S I M U L A T I O N :  
A C T I V E   D U T Y   P R O G R A M

---

This program is designed to accept numbers from the terminal for total recruits, and then use historical continuation behavior to transition the inventory. Variables used are:

ACCESS(OUTY)	This matrix is used to store the numbers input for Total Accessions for each of the outyears.
CHANGED(OUTY,RATE)	This matrix is used to store the ratings the user has changed for any of the outyears.
NUM_C(OUTY)	This matrix holds the counts for the number of ratings changed by the user for each of the outyears.
CNT	This counter is used for calculating how many zeros to DATA out in the SUM array.
ALLOC(20,RATE+1,PROG+1)	This matrix is read into the program and it stores the PRIDE allocations.
LOS_1(OUTY,RATE+1,PROG+1)	This matrix stores the original allocation of outyears total recruits before any changes have been made. This matrix is really only used in the WRITEOUT subroutine.
LOS_1_C(OUTY,RATE+1,PROG+1)	This matrix stores the original allocation of outyear recruits after the user has made changes in the rating or program mixes.
N-D(LOS-2,RATE,PROG)	This matrix is read into the program and it stores the percents used to calculate how many people to add back into a los,rating,prog cell that is growing.
TRANS(LOS-2,RATE,PROG)	This matrix is read into the program and it stores the continuation rates selected by the user to be applied to the history. It is capped at 1.0
OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)	This matrix is where all the numbers that the simulation produces are stored. The first two positions in the outyear dimension are filled with history which

```

C                                     is read into the program.
C
C      SUM(OUTY,RATE,PAYG)           This is the matrix used to compare to
C                                     requirments in the WRITEOUT procedure.
C                                     It is computed by collapsing the
C                                     OUTYEAR matrix down and adding PAYGRADE.
C-----

```

IMPLICIT NONE

```

C-----
C  The INCLUDE for Screen_Parameters contains the variables
C  used for the screen handling subroutines. They are:
C

```

```

C      BE = 'BE'   used to sound the bell
C      BL = 'BL'   used to make the screen blink
C      BO = 'BO'   used to make the screen bold
C      CL = 'CL'   used for clearing the screen
C      NE = 'NE'   used to make the screen negative
C      NO = 'NO'   used to set screen back to normal
C      SC = 'SC'   used to score (underline) data on screen
C      SK = ' '    used to skip an option.
C-----

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN\_PARAMETERS.FOR'

```

C-----
C  The INCLUDE for Parameters_Res.for contains the variables
C  to show how the Arrays should be dimensioned. They are:
C

```

```

C      LOS = 31      There are 31 LOS's in the ACTIVES
C      RATE = 69     There are 69 Ratings
C      PROG = 7      There are 7 programs in the ACTIVES
C      PAYG = 9      There are 9 paygrades
C      OUTY = 10     There are 10 outyears for this simulation
C-----

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

INTEGER ACCESS(OUTY), CHANGED(OUTY,RATE), CNT, NUM\_C(OUTY)

```

REAL ALLOC(20,RATE+1,PROG+1), LOS_1(OUTY,RATE+1,PROG+1),
•   LOS_1_C(OUTY,RATE+1,PROG+1), N_D(LOS-2,RATE,PROG),
•   OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1), SUM(OUTY,RATE,PAYG),
•   TRANS(LOS-2,RATE,PROG), LOSSES(OUTY,RATE,PROG-3)

```

PARAMETER(CNT=OUTY•RATE•PAYG)

CHARACTER\*1 HOLD

CHARACTER\*2 YR\_SEL

DATA SUM/CNT•0.0/, NUM\_C/OUTY•0/

OPEN (UNIT=6,STATUS='OLD',RECL=500)



C-----  
C  
C  
C  
C-----

M A I N L O G I C

CALL CHANGE\_SCREEN(CL,SK,SK)  
CALL DISPLAY(10,15,'A C T I V E F O R C E S I M U L A T I O N'  
                  ,BO,SK,SK)  
CALL DISPLAY(22,25,'Hit Return to Start Run',SK,SK,SK)  
CALL ACCEPT(22,49,HOLD,SK,SK,SK)  
  
CALL ACCEPT\_OUTYEAR\_RECRUITS(ALLOC,ACCESS,OUTYEAR,LOS\_1,LOS\_1\_C)  
  
CALL CHECK\_SCHOOL\_SEATS(OUTYEAR)  
  
CALL CHANGE\_MIXES(ALLOC,OUTYEAR,NUM\_C,CHANGED,LOS\_1\_C)  
  
CALL CHECK\_TO\_SEE\_IF\_SAVE\_NEW\_ALLOC(LOS\_1\_C)  
  
CALL TRANSITION(ACCESS,OUTYEAR,LOSSES)  
  
CALL ADD\_PAYGRADE(OUTYEAR,SUM)  
  
CALL WRITEIT(SUM,NUM\_C,CHANGED,YR\_SEL,LOS\_1,LOS\_1\_C,LOSSES)  
  
CALL CHANGE\_SCREEN(CL,SK,SK)  
CALL DISPLAY(10,20,'Finish of Active Force Simulation',BO,SK,SK)  
CALL DISPLAY(23,27,'Hit Return to Finish',SK,SK,SK)  
CALL ACCEPT(23,48,HOLD,SK,SK,SK)  
CALL CHANGE\_SCREEN(CL,SK,SK)  
  
STOP  
END

---

SUBROUTINE ACCEPT OUTYEAR RECRUITS

---

This subroutine is set up to allow the user to input via the terminal the total number of new recruits for each outyear (up to 20). First the user will input the accessions minus Sea College, then the user will be allowed to input for Sea College recruits. The variables used are:

ROW, I, J, K, L, START, FINISH	These variables are used as indexes into arrays, and for calculating the position the cursor on the screen.
ACCESS(OUTY)	This array stores the total recruits minus Sea College.
SEA_C(OUTY)	This array stores the total Sea College recruits input for each outyear.
FLAG	This variable is used to flag whether the user is inputting for Sea College or for all other recruits.
INT, MA	Used as integer number holders.
DIS_YEAR	Used for storing year to be displayed on the terminal so the user will know which year is being worked with.
ALLOC(20, RATE+1, PROG+1)	This matrix is read into the program in this subroutine and it stores how the LOS 1 recruits are to be broken out by rating and program.
OUTYEAR(OUTY+2, LOS, RATE+1, PROG+1)	In this subroutine Outyear will be filled for LOS 1 in each of the outyears. Also the first Two outyear dimensions will be filled with HISTORY which will be read in this subroutine.
HIST_1(LOS, RATE+1, PROG+1)	This matrix will store the year-1 history and is read into the program.
HIST_2(LOS, RATE+1, PROG+1)	This matrix will store the year-2 history and is read into the program. HIST_1, HIST_2 will be loaded into OUTYEAR.
MAX(20, RATE+1, 9)	This matrix is read into the program and stores the maximum number allowed

```

C                                     for each rating for each of the
C                                     outyears. This matrix is used to flag
C                                     for the user the ratings which exceed
C                                     the school plan.
C
C      LOS_1(OUTY,RATE+1,PROG+1)      This matrix will store the original
C                                     LOS 1 breakout after the user inputs
C                                     the total recruits and they are
C                                     allocated.
C
C      LOS_1_C(OUTY,RATE+1,PROG+1)    This matrix will store the original
C                                     LOS 1 breakout and any changes that
C                                     have been made. Equals LOS_1 after
C                                     this subroutine because no changes
C                                     yet.
C
C-----

```

```

      SUBROUTINE ACCEPT_OUTYEAR_RECRUITS(ALLOC,ACCESS,OUTYEAR,LOS_1,
      *                               LOS_1_C)

      IMPLICIT NONE

      INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
      INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

      INTEGER ROW, I, J, K, L, START, FINISH, ACCESS(OUTY),
      *      SEA_C(OUTY), FLAG, INT, MA, DIS_YEAR, ANS

      REAL ALLOC(20,RATE+1,PROG+1), OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1),
      *      MAX(20,RATE+1,9), HIST_2(LOS,RATE+1,PROG+1),
      *      HIST_1(LOS,RATE+1,PROG+1), LOS_1(OUTY,RATE+1,PROG+1),
      *      LOS_1_C(OUTY,RATE+1,PROG+1)

      CHARACTER*2 HOLD

      CHARACTER*8 RATE_LABEL

      CHARACTER*50 FILEN
      CHARACTER*25 NAME

```

```

C-----
C  First thing to do is select the allocation matrix and
C  to read in the two history files, then load the history into OUTYEAR.

```

```

      OPEN(UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
      *      FILE='CNA2:[CORLISSG.FORCE.DAT]H_2.DAT')
      READ(59) HIST_2
      CLOSE(UNIT=59)

      OPEN(UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
      *      FILE='CNA2:[CORLISSG.FORCE.DAT]HIST.DAT')
      READ(59) HIST_1
      CLOSE(UNIT=59)

      DO 10 L=1,PROG

```

```

        DO 10 K=1,RATE
            DO 10 J=1,LOS
                OUTYEAR(1,J,K,L) = HIST_2(J,K,L)
10          OUTYEAR(2,J,K,L) = HIST_1(J,K,L)

C-----
20      CONTINUE

        CALL CHANGE_SCREEN(CL,SK,SK)

        CALL DISPLAY(10,15,'1',BO,SK,SK)
        CALL DISPLAY(10,16,' - Use the default Allocation matrix',
            SK,SK,SK)
        CALL DISPLAY(12,15,'2',BO,SK,SK)
        CALL DISPLAY(12,16,' - Select a user created Allocation matrix',
            SK,SK,SK)
        CALL DISPLAY(15,20,'Enter your selection:',SK,SK,SK)

30      CONTINUE

        CALL ACCEPT_INTEGER(15,42,ANS,1,BO,SC,SK)

C-----
C   If the default is used then assign the default filename to FILEN
C   But if the user wants to select an allocation he/she created then
C   call FIND_FILE and the filename will be passed back.

        IF(ANS .NE. 1 .AND. ANS .NE. 2) THEN
            CALL DISPLAY(15,45,'Invalid input <Hit Return>',BO,NE,BE)
            CALL ACCEPT(15,71,HOLD,SK,SK,SK)
            CALL DISPLAY(15,45,'',SK,SK,SK)
            GOTO 30
        ELSE IF (ANS .EQ. 1) THEN
            FILEN = 'CNA2:[CORLISSG.FORCE.DAT]NEW_ALLOC.DAT'
        ELSE
            CALL FIND_FILE(FILEN)
        END IF

C-----
C   FIND_FILE subroutine will pass back 'FLAG' in the FILEN if there were
C   no user created allocation to select from. If this happens the user
C   must select the default or will not pass this point.

        IF (FILEN .EQ. 'FLAG') THEN
            GOTO 20
        END IF

C-----
C   Next open the file selected.
        OPEN (UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',FILE=FILEN)
        READ(59) ALLOC
        CLOSE (UNIT=59)

C-----
C   The FLAG is used to show wether inputing for Sea College or not.
        FLAG = 0

```

```

40      CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      IF (FLAG .EQ. 0) THEN
          CALL DISPLAY(6,1,'Total Accessions',BO,NE,BL)
          CALL DISPLAY(7,1,'Minus Sea College',BO,NE,BL)
      ELSE
          CALL DISPLAY(6,1,'Inputs for',BO,NE,BL)
          CALL DISPLAY(7,1,'Sea College',BO,NE,BL)
      END IF

      DIS_YEAR = CFY

      ROW = 2
      I = 1

      CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
      CALL DISPLAY(ROW,21,' - Current FY',SK,SK,SK)
      CALL DISPLAY(ROW,34,' Total Recruits      : ',SK,SK,SK)
      CALL DISPLAY_INTEGER(ROW,50,DIS_YEAR,4,SK,SK,SK)
      CALL DISPLAY(ROW,56,'                ',BO,SC,SK)

      DO 50 I=2,OUTY
          DIS_YEAR = DIS_YEAR + 1

          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF
          CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
          CALL DISPLAY(ROW,21,' - Outyear',SK,SK,SK)
          CALL DISPLAY_INTEGER(ROW,32,I-1,2,SK,SK,SK)
          CALL DISPLAY(ROW,34,' Total Recruits      : ',SK,SK,SK)
          CALL DISPLAY_INTEGER(ROW,50,DIS_YEAR,4,SK,SK,SK)
          CALL DISPLAY(ROW,56,'                ',BO,SC,SK)
50      CONTINUE

      IF (OUTY .GT. 10) THEN
          ROW = 1
      ELSE
          ROW = 0
      END IF

      START = 1
      FINISH = OUTY

60      CONTINUE

      DO 70 I=START,FINISH
          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF

```

```

        IF (FLAG .EQ. 0) THEN
            ACCESS(I) = 0
            CALL ACCEPT_INTEGER(ROW,56,ACCESS(I),6,B0,SC,SK)
        ELSE
            SEA_C(I) = 0
            CALL ACCEPT_INTEGER(ROW,56,SEA_C(I),6,B0,SC,SK)
        END IF
70    CONTINUE

    CALL DISPLAY(22,14,'Enter Number of Outyear to change or <Return>'
    ,SK,SK,SK)

    CALL ACCEPT_INTEGER(22,60,START,2,B0,SC,SK)

    IF ((START .GT. 0) .AND. (START .LE. OUTY)) THEN
        IF (OUTY .GT. 10) THEN
            ROW = START
        ELSE
            ROW = (START - 1) * 2
        END IF

        FINISH = START

        GOTO 60
    END IF

C   Set FLAG = 1 for inputing Sea College
    IF (FLAG .EQ. 0) THEN
        FLAG = 1
        GOTO 40
    END IF

C-----
C   The 80 loops will take the inputted accessions and allocate
C   them into OUTYEAR using ALLOC.
    DO 80 K=1,PROG+1
        DO 80 J=1,RATE+1
            DO 80 I=3,OUTY+2
80      OUTYEAR(I,1,J,K) = ACCESS(I-2) * ALLOC(I-2,J,K)
C-----

C-----
C   The 90 loop is used to update ACCESS, OUTYEAR matrixes with
C   the new program Sea College.
    DO 90 I=3,OUTY+2
        ACCESS(I-2) = ACCESS(I-2) + SEA_C(I-2)

        OUTYEAR(I,1,1,6) = SEA_C(I-2) * 0.27
        OUTYEAR(I,1,32,6) = SEA_C(I-2) * 0.29
        OUTYEAR(I,1,61,6) = SEA_C(I-2) * 0.44

        OUTYEAR(I,1,1,8) = OUTYEAR(I,1,1,8) + OUTYEAR(I,1,1,6)
        OUTYEAR(I,1,32,8) = OUTYEAR(I,1,32,8) + OUTYEAR(I,1,32,6)
        OUTYEAR(I,1,61,8) = OUTYEAR(I,1,61,8) + OUTYEAR(I,1,61,6)

        OUTYEAR(I,1,70,6) = OUTYEAR(I,1,70,6) + SEA_C(I-2)

```

90                   OUTYEAR(I,1,70,8) = OUTYEAR(I,1,70,8) + SEA\_C(I-2)

C-----  
C   The 100 loop is used to fill up LOS\_1 and LOS\_1\_C with OUTYEAR's  
C   LOS 1 for each of the outyears.

      DO 100 I=1,OUTY  
          DO 100 J=1,RATE+1  
              DO 100 K=1,PROG+1  
                  LOS\_1(I,J,K) = OUTYEAR(I+2,1,J,K)  
100               LOS\_1\_C(I,J,K) = OUTYEAR(I+2,1,J,K)

C-----

      RETURN  
      END

```

C-----
C
C      S U B R O U T I N E   F I N D   F I L E
C
C-----
C      This subroutine is used to allow the user to select any previously
C      created and saved allocation matrix.  The subroutine searches the
C      users directory for allocation files and displays to the terminal the
C      choices.  If no choices are available, 'FLAG' is put in FILEN.
C
C      FILEN                                This is passed back to the main
C                                           program and it contains the filename
C                                           of the users choice or 'FLAG'.
C
C      NEWFILE                             This is used in the library call and
C                                           it contains the next file found in the
C                                           directory.  When no more files it
C                                           passes back the default filename.
C
C      DEFAULT, RELATED, FILENAME          These are used by the library and are
C                                           used to locate a certain set of files.
C
C      STORE(100)                          This is used to store the names of all
C                                           the user created allocations found
C                                           in the directory.
C
C      DIS(100)                            This stores the names of the files to
C                                           be displayed to the user on the screen.
C
C      CHECK                               Is used to check and see when there are
C                                           no more files.
C
C      CONTEXT                             Is used by the library subroutine as an
C                                           address pointer and must be set to 0
C                                           at start.
C
C      ROW, ROW1                           Used to calculate which row of the
C                                           screen to display on.
C
C      BEGIN, END                          Used to store the begin and end of
C                                           strings that are being searched for in
C                                           other strings.
C
C      NUMFILES                            Is a counter used to keep count of the
C                                           of allocations found in the directory.
C
C      ANS                                 The answer to which allocation the user
C                                           selected to use.
C
C      I, DIFF                             Used as integers for loops and
C                                           calculations.
C-----

```

```

SUBROUTINE FIND_FILE(FILEN)

```

```

IMPLICIT NONE

```



```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

INTEGER CONTEXT, ROW, ROW1, BEGIN, END, NUMFILES, ANS, I, DIFF

CHARACTER HOLD

CHARACTER*50 FILENAME, NEWFILE, DEFAULT, RELATED, FILEN,
      STORE(100), DIS(100), CHECK

CALL CHANGE_SCREEN(CL,SK,SK)

FILENAME = 'ZZZ_*.DAT'

DEFAULT = FILENAME
RELATED = FILENAME

CONTEXT = 0
NUMFILES = 1

CHECK = '

10 CONTINUE

CALL LIB$FIND_FILE(FILENAME,NEWFILE,CONTEXT,DEFAULT,RELATED)

BEGIN = INDEX(NEWFILE,']ZZZ_')
END = INDEX(NEWFILE,'.DAT')
BEGIN = BEGIN + 1
END = END + 3
DIFF = (END - BEGIN) + 1

CHECK(1:DIFF) = NEWFILE(BEGIN:END)
DIS(NUMFILES) = CHECK(5:DIFF-4)

IF (CHECK .NE. FILENAME) THEN
  IF (NEWFILE .NE. STORE(NUMFILES)) THEN
    STORE(NUMFILES) = NEWFILE
    NUMFILES = NUMFILES + 1
  END IF
  CHECK = '
  GOTO 10
END IF

NUMFILES = NUMFILES - 1

IF (NUMFILES .EQ. 0) THEN
  CALL DISPLAY(10,10,'No user created allocations yet <Hit Return>'
    ,80,BE,SK)
  CALL ACCEPT(10,55,HOLD,SK,SK,SK)
  FILEN = 'FLAG'
  GOTO 999
END IF

ROW = 0
ROW1 = 0

```

```

DO 20 I=1,NUMFILES
    IF (NUMFILES .GT. 10) THEN
        ROW = ROW + 1
    ELSE
        ROW = ROW + 2
    END IF

    IF (I .GT. 20) THEN
        ROW1 = ROW1 + 1
        CALL DISPLAY_INTEGER(ROW1,46,I,3,B0,SK,SK)
        CALL DISPLAY(ROW1,50,'--',SK,SK,SK)
        CALL DISPLAY(ROW1,52,DIS(I),SK,SK,SK)
    ELSE
        CALL DISPLAY_INTEGER(ROW,10,I,3,B0,SK,SK)
        CALL DISPLAY(ROW,14,'--',SK,SK,SK)
        CALL DISPLAY(ROW,16,DIS(I),SK,SK,SK)
    END IF

20  CONTINUE

    CALL DISPLAY(23,10,'Enter your selection:',SK,SK,SK)

30  CONTINUE
    CALL ACCEPT_INTEGER(23,32,ANS,2,B0,SC,SK)

    IF (ANS .LT. 1 .OR. ANS .GT. NUMFILES) THEN
        CALL DISPLAY(23,35,'Invalid input <Hit Return>',B0,NE,BE)
        CALL ACCEPT(23,62,HOLD,SK,SK,SK)
        CALL DISPLAY(23,35,'                                ',SK,SK,SK)
        GOTO 30
    END IF

    FILEN = STORE(ANS)

999 CONTINUE

    RETURN
    END

```

---

S U B R O U T I N E   C H E C K   S C H O O L   S E A T S

---

This subroutine is used to display to the user how the newly allocated new recruits are different from the planned school seat allocations.

OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)      This matrix has the Two history dimensions (1,2) of outyear filled in and the LOS 1 dimension filled in for each outyear.

MAX(20,RATE+1,9)      This matrix is read into the program in this subroutine and it contains the School plans for FY 1988.

ROW, COLUMN      Used to calc. where to display the number on the screen.

I,J,K,L      Used in loops.

START, FINISH      Used to show which ratings to display at the screen.

INT, MA, DIFF, R      Used as integers for calculating and storing numbers.

---

SUBROUTINE CHECK\_SCHOOL\_SEATS(OUTYEAR)

IMPLICIT NONE

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN\_PARAMETERS.FOR'

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

INTEGER ROW, COLUMN, I, J, K, L, INT, MA, START, FINISH,  
DIFF, R

REAL OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1), MAX(20,RATE+1,9)

CHARACTER\*1 HOLD, ANS

CHARACTER\*8 RATE\_LABEL

CHARACTER\*50 FILEN

CHARACTER\*25 NAME

CALL CHANGE\_SCREEN(CL,SK,SK)

CALL DISPLAY(12,10,'Would you like to view the comparison of',

```

      SK,SK,SK)
      CALL DISPLAY(13,10,'Allocations to Planned School Seats (',
      SK,SK,SK)
      CALL DISPLAY(13,47,'Y',80,SK,SK)
      CALL DISPLAY(13,48,'es or',SK,SK,SK)
      CALL DISPLAY(13,54,'N',80,SK,SK)
      CALL DISPLAY(13,55,'o)?',SK,SK,SK)
5      CONTINUE
      CALL ACCEPT(13,59,ANS,80,SC,SK)
      IF (ANS .NE. 'N' .AND. ANS .NE. 'Y' .AND. ANS .NE. 'n' .AND.
      ANS .NE. 'y') THEN
          GOTO 5
      END IF

      IF (ANS .EQ. 'N' .OR. ANS .EQ. 'n') THEN
          GOTO 999
      END IF

      CALL CHANGE_SCREEN(CL,SK,SK)

```

```

C-----
C The OUTYEAR matrix for LOS 1, the LOS 1 is
C compared to the MAX array to see if it exceeds the school plan.
C First read in MAX.

```

```

      OPEN (UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
      FILE='CNA2:[CORLISSG.FORCE.DAT]MAX.DAT')
      READ (59) MAX
      CLOSE (UNIT=59)

      ROW = 0
      COLUMN = 4

      START = 1
      FINISH = RATE

      DO 10 I=START,FINISH
          ROW = ROW + 1
          IF (I .EQ. 24 .OR. I .EQ. 47) THEN
              ROW = 1
              COLUMN = COLUMN + 25
          END IF
          INT = OUTYEAR(5,1,I,8) + .5
          MA = MAX(3,I,8)
          DIFF = INT - MA
          R = (DIFF / MAX(3,I,8)) * 100
          IF (MA .LT. INT) THEN
              CALL DISPLAY(ROW,COLUMN,'
              NE,SK,SK)
              CALL DISPLAY_INTEGER(ROW,COLUMN,DIFF,7,80,NE,SK)
              CALL DISPLAY_INTEGER(ROW,COLUMN+8,R,3,80,NE,SK)
              CALL DISPLAY(ROW,COLUMN+11,'% ',80,NE,SK)
              CALL DISPLAY(ROW,COLUMN+14,RATE_LABEL(I),NE,SK,SK)
          ELSE IF (MA .EQ. 999999) THEN
              CALL DISPLAY(ROW,COLUMN,'No School Req',80,SK,SK)
              CALL DISPLAY(ROW,COLUMN+14,RATE_LABEL(I),SK,SK,SK)
          
```

```

        ELSE
            CALL DISPLAY_INTEGER(ROW,COLUMN,DIFF,7,B0,SK,SK)
            CALL DISPLAY_INTEGER(ROW,COLUMN+8,R,3,B0,SK,SK)
            CALL DISPLAY(ROW,COLUMN+11,'% ',B0,SK,SK)
            CALL DISPLAY(ROW,COLUMN+14,RATE_LABEL(1),SK,SK,SK)
        END IF
10      CONTINUE

        CALL ACCEPT(23,52,HOLD,SK,SK,SK)

999    CONTINUE

        CALL CHANGE_SCREEN(CL,SK,SK)

        RETURN
    END

```

```

C-----
C
C      S U B R O U T I N E   C H A N G E   M I X E S
C
C-----
C
C      This subroutine is used to control which Outyear the user wants to
C      change, and wether to change Rating or Program mixes for LOS 1.
C
C      I, J, K, ROW      These variables are used as indexes
C                        into arrays and pointers to positions
C                        on the screen for displaying.
C
C      YEAR              This variable holds the year the user
C                        selected to work on.
C
C      DIS_YEAR          Is used to display to the terminal
C                        which year the user is dealing with.
C
C      ANSWER            Will store the answer as to whether
C                        the user wants to change Program or
C                        Rating mixes.
C
C      DONE(OUTY)        This is an array of flags used to
C                        flag which years the user has changed.
C
C      NUM_C(OUTY)       This matrix has the counts for the
C                        total number of Rating changes made
C                        for each outyear.
C
C      CHANGED(OUTY,RATE) This matrix has the ratings stored that
C                        the user has changed for each outyear.
C
C      ALLOC(20,RATE+1,PROG+1) Has the allocation matrix for each
C                        outyear.
C
C      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1) Has the Two history years and LOS 1
C                        for each of the outyears with any
C                        changes made to LOS 1.
C
C      LOS_1_C(OUTY,RATE+1,PROG+1) This matrix stores the LOS 1 allocated
C                        new recruits with any changes that have
C                        been made.
C-----

```

```

SUBROUTINE CHANGE_MIXES(ALLOC,OUTYEAR,NUM_C,CHANGED,LOS_1_C)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

INTEGER I, J, K, ANSWER, YEAR, DIS_YEAR, ROW, NUM_C(OUTY),
•      CHANGED(OUTY,RATE), DONE(OUTY)

```

```

REAL ALLOC(20,RATE+1,PROG+1), OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1),

```

• LOS\_1\_C(OUTY,RATE+1,PROG+1)

CHARACTER HOLD, ANS

```
C-----
C First check to see if any changes are required.
  CALL CHANGE_SCREEN(CL,SK,SK)
  CALL DISPLAY(13,2,'Would you like to change Program',SK,SK,SK)
  CALL DISPLAY(13,35,'or Rating mixes for any year (' ,SK,SK,SK)
  CALL DISPLAY(13,65,'Y',BO,SK,SK)
  CALL DISPLAY(13,66,'es or',SK,SK,SK)
  CALL DISPLAY(13,72,'N',BO,SK,SK)
  CALL DISPLAY(13,73,'o)?',SK,SK,SK)
5  CONTINUE
  CALL ACCEPT(13,77,ANS,BO,SC,SK)
  IF (ANS .NE. 'N' .AND. ANS .NE. 'Y' .AND. ANS .NE. 'n' .AND.
    • ANS .NE. 'y') THEN
    GOTO 5
  END IF

C If no changes then exit this subroutine.

  IF (ANS .EQ. 'N' .OR. ANS .EQ. 'n') THEN
    GOTO 9999
  END IF

C-----

10  CONTINUE

  CALL CHANGE_SCREEN(CL,SK,SK)

  ROW = 2
  I = 1

  DIS_YEAR = CFY

  CALL DISPLAY_INTEGER(ROW,24,I,2,BO,SK,SK)
  CALL DISPLAY(ROW,27,'- Current FY',SK,SK,SK)
  CALL DISPLAY_INTEGER(ROW,40,DIS_YEAR,4,SK,SK,SK)
  IF (DONE(1) .EQ. 1) THEN
    CALL DISPLAY(ROW,21,'X',BO,NE,SK)
  END IF

  DO 20 I=2,OUTY
    DIS_YEAR = DIS_YEAR + 1

    IF (OUTY .LE. 10) THEN
      ROW = ROW + 2
    ELSE
      ROW = ROW + 1
    END IF
    CALL DISPLAY_INTEGER(ROW,24,I,2,BO,SK,SK)
    CALL DISPLAY(ROW,27,'- Outyear',SK,SK,SK)
    CALL DISPLAY_INTEGER(ROW,37,I-1,2,SK,SK,SK)
    CALL DISPLAY_INTEGER(ROW,40,DIS_YEAR,4,SK,SK,SK)
    IF (DONE(1) .EQ. 1) THEN
```

```

                CALL DISPLAY(ROW,21,'X',BO,NE,SK)
            END IF
20      CONTINUE

        CALL DISPLAY(22,20,'Enter Your Selection or (',SK,SK,SK)
        CALL DISPLAY(22,45,'99',BO,SK,SK)
        CALL DISPLAY(22,48,'to end):',SK,SK,SK)

30      CONTINUE

C-----
C  Accept the year and check to see if good
C  selection or finished.
        CALL ACCEPT_INTEGER(22,57,YEAR,2,BO,SC,SK)

        IF (YEAR .EQ. 99) THEN
            GOTO 999
        ELSE IF (YEAR .LT. 1 .OR. YEAR .GT. OUTY) THEN
            CALL DISPLAY(22,60,'Bad input Hit Return',BO,NE,BE)
            CALL ACCEPT(22,80,HOLD,SK,SK,SK)
            CALL DISPLAY(22,60,'',SK,SK,SK)
            GOTO 30
        END IF
C-----

        DIS_YEAR = YEAR + (CFY - 1)

        CALL CHANGE_SCREEN(CL,SK,SK)

40      CONTINUE

        CALL DISPLAY(7,20,'Selection for year',SC,SK,SK)
        CALL DISPLAY_INTEGER(7,39,DIS_YEAR,4,BO,SK,SK)
        CALL DISPLAY(10,20,'1',BO,SK,SK)
        CALL DISPLAY(10,21,' - Change Program Mixes',SK,SK,SK)
        CALL DISPLAY(12,20,'2',BO,SK,SK)
        CALL DISPLAY(12,21,' - Change Rating Mixes',SK,SK,SK)
        CALL DISPLAY(14,20,'9',BO,SK,SK)
        CALL DISPLAY(14,21,' - End Changing this Outyear',SK,SK,SK)

        CALL DISPLAY(17,20,'Enter Your Selection:',SK,SK,SK)

50      CONTINUE

C-----
C  Choose whether to change Rating or Program mixes and
C  continue until 9 is entered. After 9 is entered go
C  back to selecting a new year to work with.
        CALL ACCEPT_INTEGER(17,42,ANSWER,1,BO,SC,SK)

        IF (ANSWER .EQ. 1) THEN
            CALL CHANGE_PROGRAM(YEAR,ALLOC,OUTYEAR,DONE)
            GOTO 40
        ELSE IF (ANSWER .EQ. 2) THEN
            CALL CHANGE_RATING(YEAR,ALLOC,OUTYEAR,NUM_C,CHANGED,DONE)
            GOTO 40

```



```

        ELSE IF (ANSWER .EQ. 9) THEN
            GOTO 10
        END IF
C-----
C  If you get here then inputed bad selection so display error
C  and go back.
        CALL DISPLAY(17,44,'Bad selection <Hit Return>',BO,NE,BE)
        CALL ACCEPT(17,71,HOLD,SK,SK,SK)
        CALL DISPLAY(17,44,'',SK,SK,SK)

        GOTO 50

999      CONTINUE
C-----
C-----
C  When finished all changes then update LOS_1_C matrix.
        DO 60 I=1,OUTY
            DO 60 J=1,RATE+1
                DO 60 K=1,PROG+1
50          LOS_1_C(I,J,K) = OUTYEAR(I+2,1,J,K)

9999     CONTINUE

        RETURN
        END

```

---

S U B R O U T I N E   C H A N G E   P R O G R A M

---

This subroutine is used to change the program mixes for a certain outyear. The total number of accessions for the outyear must be the same as entering before the user is allowed to exit.

I, J, K, START, FINISH, ROW      These variables are used as indexes into arrays and as pointers to positions to display on the screen.

YEAR, YR      These are used to show which year the user has selected to work with and what position that is in OUTYEAR.

CNT      Is used to count the number of changes made.

NUM, DIFF      These are used as integer numbers for displaying and computing.

ORIG(PROG+1)      Used to store the rounded integer number of how many were originally allocated to each Program. Used for display purposes only.

PROGRAMB(PROG+1)      This matrix is the same as ORIG except the changes that the user makes are done to this matrix. Used for computing and displaying.

D(PROG+1)      This matrix stores the differences of the changes made to Program mixes. Used for computing and displaying.

DONE(OUTY)      If a change is made to this outyear a 1 is placed there to flag it.

ALLOC(20,RATE+1,PROG+1)      Has the allocation matrix from each outyear.

OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)      Has the two histories and the LOS 1 for each of the outyears. Will have the Program changes after exit.

REALNUM, DIFFR      These are real number uses for computations.

---

SUBROUTINE CHANGE\_PROGRAM(YR,ALLOC,OUTYEAR,DONE)

IMPLICIT NONE

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

      INTEGER YEAR, YR, START, FINISH, I, J, K, ORIG(PROG+1),
      *      CNT, DIFF, ROW, PROGRAMB(PROG+1), D(PROG+1),
      *      NUM, DONE(OUTY)

```

```

      REAL ALLOC(20,RATE+1,PROG+1), OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1),
      *      REALNUM, DIFFR

```

```

CHARACTER HOLD

```

```

YEAR = YR + 2

```

```

CNT = 0
START = 1
FINISH = PROG+1
ROW = 2

```

```

C-----

```

```

C The first thing to do is store what the LOS 1
C looks like by Program for the selected outyear.
C The 10 loop loads in LOS 1 and computes a total accession.

```

```

      ORIG(PROG+1) = 0
      PROGRAMB(PROG+1) = 0
      D(PROG+1) = 0

```

```

DO 10 I=1,PROG
      ORIG(I) = 0
      PROGRAMB(I) = 0
      D(I) = 0
      ORIG(I) = OUTYEAR(YEAR,1,RATE+1,I) + .5
      PROGRAMB(I) = OUTYEAR(YEAR,1,RATE+1,I) + .5
      ORIG(PROG+1) = ORIG(PROG+1) + ORIG(I)
10    PROGRAMB(PROG+1) = PROGRAMB(PROG+1) + PROGRAMB(I)

```

```

C-----

```

```

CALL CHANGE_SCREEN(CL,SK,SK)

```

```

CALL DISPLAY (2,39,'Orig',SC,SK,SK)
CALL DISPLAY (2,50,'Change',SC,SK,SK)
CALL DISPLAY (2,62,'Diff',SC,SK,SK)

```

```

CALL DISPLAY (4,10,'1',BO,SK,SK)
CALL DISPLAY (4,11,' - 4YO Program',SK,SK,SK)
CALL DISPLAY (6,10,'2',BO,SK,SK)
CALL DISPLAY (6,11,' - Active Mariner Program',SK,SK,SK)
CALL DISPLAY (8,10,'3',BO,SK,SK)
CALL DISPLAY (8,11,' - 5&6YO Program',SK,SK,SK)
CALL DISPLAY(10,10,'4',BO,SK,SK)
CALL DISPLAY(10,11,' - Prior Service Program',SK,SK,SK)
CALL DISPLAY(12,10,'5',BO,SK,SK)
CALL DISPLAY(12,11,' - TAR Program',SK,SK,SK)
CALL DISPLAY(14,10,'6',BO,SK,SK)
CALL DISPLAY(14,11,' - Sea College Program',SK,SK,SK)
CALL DISPLAY(16,10,'7',BO,SK,SK)

```

```
CALL DISPLAY(16,11,' - Any new program      ',SK,SK,SK)
CALL DISPLAY(18,28,'Totals',SK,SK,SK)
```

```
DO 20 I=START,FINISH
    ROW = ROW + 2
    CALL DISPLAY_INTEGER(ROW,37,ORIG(I),6,BO,SK,SK)
    CALL DISPLAY_INTEGER(ROW,50,PROGRAMB(I),6,BO,SC,SK)
    CALL DISPLAY_INTEGER(ROW,60,D(I),6,BO,SK,SK)
```

```
20 CONTINUE
```

```
CALL DISPLAY(21,10,'Enter Program to change or <Return>:',SK,SK,SK)
```

```
30 CONTINUE
CALL ACCEPT_INTEGER(21,47,START,1,BO,SC,SK)
```

```
C-----
```

```
C After a program is selected to change the user inputs
C the new number and the difference is computed and displayed.
C If a change is made an equal change must be made in some other
C Program to offset it.
```

```
IF (START .GT. 0 .AND. START .LT. PROG+1) THEN
    IF (START .EQ. 6) THEN
        CALL DISPLAY(23,15,'Cannot change Sea College here',
                     BO,NE,BE)
        CALL DISPLAY(23,46,'<Hit Return>',SK,SK,SK)
        CALL ACCEPT(23,58,HOLD,SK,SK,SK)
        CALL DISPLAY(23,15,'
                               SK,SK,SK)
        CALL DISPLAY(23,46,'
                               ',SK,SK,SK)
        GOTO 30
    END IF
    CNT = CNT + 1
    ROW = 2 + (START * 2)
    NUM = PROGRAMB(START)
    CALL ACCEPT_INTEGER(ROW,50,PROGRAMB(START),6,BO,SC,SK)
    D(START) = D(START) + (PROGRAMB(START) - NUM)
    D(PROG+1) = D(PROG+1) + (PROGRAMB(START) - NUM)
    PROGRAMB(PROG+1) = PROGRAMB(PROG+1) + (PROGRAMB(START) - NUM)
    CALL DISPLAY_INTEGER(ROW,60,D(START),6,BO,SK,NE)
    CALL DISPLAY_INTEGER(18,60,D(PROG+1),6,BO,SK,NE)
    CALL DISPLAY_INTEGER(18,50,PROGRAMB(PROG+1),6,BO,SK,NE)
    GOTO 30
```

```
END IF
```

```
C-----
```

```
C If no changes were made then exit subroutine.
```

```
IF (CNT .EQ. 0) THEN
    GOTO 999
```

```
END IF
```

```
DONE(YEAR-2) = 1
```

```
DIFF = PROGRAMB(PROG+1) - ORIG(PROG+1)
```

```
C-----
```

```
C This IF statement is checking to make sure that before
```

C exiting the total is the same as when entering the subroutine.  
 C If not an error message is sent to the user and more changes  
 C must be made.

```

    IF (DIFF .GT. 0 .OR. DIFF .LT. 0) THEN
      CALL DISPLAY(23,10,'The Total has to =',BO,NE,BE)
      CALL DISPLAY_INTEGER(23,29,ORIG(PROG+1),6,BO,SC,SK)
      CALL DISPLAY(23,36,'not',BO,NE,SK)
      CALL DISPLAY_INTEGER(23,40,PROGRAMB(PROG+1),6,BO,SC,SK)
      CALL DISPLAY(23,49,'<Hit Return>',BO,NE,SK)
      CALL ACCEPT(23,62,HOLD,SK,SK,SK)
      CALL DISPLAY(23,10,'
      *                               ,SK,SK,SK)
      CALL DISPLAY(23,45,'
      *                               ,SK,SK,SK)

      GOTO 30
    END IF

```

C-----  
 C After all changes are made and the totals equal the excess or  
 C deficit in each new Program total must be distributed back over all  
 C the Ratings using the ALLOC matrix. The 40 loops do this.

```

    DO 40 K=1,PROG
      DIFFR = PROGRAMB(K) - OUTYEAR(YEAR,1,RATE+1,K)
      DO 40 J=1,RATE
        IF (ALLOC(YEAR,RATE+1,K) .NE. 0) THEN
          OUTYEAR(YEAR,1,J,K) = OUTYEAR(YEAR,1,J,K) +
          * ((ALLOC(YEAR,J,K)/ALLOC(YEAR,RATE+1,K)) * DIFFR)
        END IF

```

40 CONTINUE

C-----  
 C The 50 and 60 loops zero out the Program and Rating total so they  
 C can be recomputed.

```

    DO 50 K=1,PROG+1
      OUTYEAR(YEAR,1,RATE+1,K) = 0.0

      DO 60 J=1,RATE
        OUTYEAR(YEAR,1,J,PROG+1) = 0.0

```

C-----  
 C The 70 loops recompute the LOS 1 totals for Rating and Program.

```

    DO 70 J=1,RATE
      DO 70 K=1,PROG
        OUTYEAR(YEAR,1,J,PROG+1) = OUTYEAR(YEAR,1,J,PROG+1) +
        * OUTYEAR(YEAR,1,J,K)
        OUTYEAR(YEAR,1,RATE+1,K) = OUTYEAR(YEAR,1,RATE+1,K) +
        * OUTYEAR(YEAR,1,J,K)
        OUTYEAR(YEAR,1,RATE+1,PROG+1) = OUTYEAR(YEAR,1,J,K) +
        * OUTYEAR(YEAR,1,RATE+1,PROG+1)

```

70 CONTINUE

999 CONTINUE

CALL CHANGE\_SCREEN(CL,SK,SK)

RETURN

END

---

S U B R O U T I N E   C H A N G E   R A T I N G

---

This subroutine is used to change Rating mixes for the outyear the user selected. The total number for the ratings must be the same after the changes are done.

I, J, K, L, ROW, COLUMN      These variables are used as indexes into arrays and as pointers to positions on the terminal.

YEAR, YR      These are used to show which year the user has selected to work with and what position that is in OUTYEAR.

RATING\_INDEX(RATE)      This matrix stores the ratings that the user has chosen to work with.

NUM\_RATING      This is a count of the number of ratings that have been selected. <= 9.

SUMB, SUMA      These are used to show the sum of the ratings before changes are made and after. These two should equal at the end of this subroutine.

CNT      This counts the number of rating groups the user changed.

DIFF, MA      Used as integer numbers for computing and storage.

RATINGB(RATE,PROG+1)      This matrix has the allocated numbers stored for the rating selected and is used for making changes by the user. It is integer for displaying purposes.

DONE(OUTY)      This is an array of flags. If any rating changes are made in this year it is flagged 1 in that pos. of DONE.

NUM\_C(OUTY)      This matrix has the counts for the total number of Rating changes made for each outyear.

CHANGED(OUTY,RATE)      This matrix has the ratings stored that the user has changed for this outyear.

DIFFR      This is just used in the program as a real number for computations.

ALLOC(20,RATE+1,PROG+1)      Has the allocation matrix for each

```

C                                outyear.
C
C      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)  Has the two histories and the LOS 1
C                                           for each of the outyears.  Will
C                                           have the Rating changes for this
C                                           outyear after exit.
C
C      MAX(20,RATE+1,9)                This matrix is read into the program
C                                           and stores the maximum number allowed
C                                           for each rating for each of the
C                                           outyears. This matrix is used to flag
C                                           for the user the ratings which exceed
C                                           the school plan.
C-----

```

```

SUBROUTINE CHANGE_RATING(YR,ALLOC,OUTYEAR,NUM_C,CHANGED,DONE)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

INTEGER RATING_INDEX(RATE), NUM_RATING, I, J, K, SUMB, SUMA,
*      YEAR, YR, ROW, COLUMN, DIFF, RATINGB(RATE,PROG+1), CNT,
*      NUM_C(OUTY), CHANGED(OUTY,RATE), L, MA, DONE(OUTY)

```

```

REAL DIFFR, ALLOC(20,RATE+1,PROG+1), MAX(20,RATE+1,9),
*      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)

```

```

CHARACTER HOLD

```

```

CHARACTER*8 RATE_LABEL

```

```

YEAR = YR + 2

```

```

CALL CHANGE_SCREEN(CL,SK,SK)

```

```

C-----
C  Calling this subroutine allows the user to select
C  the ratings to work with.
C      CALL CHOOSE_RATING(RATING_INDEX, NUM_RATING)
C-----

```

```

C-----
C  Only 9 ratings can be worked with at one time.
C      IF (NUM_RATING .GT. 9) THEN
C          NUM_RATING = 9
C      END IF
C-----

```

```

*      OPEN(UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
*          FILE='CNA2:[CORLISSG.FORCE.DAT]MAX.DAT')
*      READ(59) MAX
*      CLOSE(UNIT=59)

```

```

CALL DISPLAY(2,13,'4Y0' ,SC,SK,SK)

```



```

CALL DISPLAY(2,18,'ACT MA',SC,SK,SK)
CALL DISPLAY(2,27,'5&6YO',SC,SK,SK)
CALL DISPLAY(2,34,'PR SER',SC,SK,SK)
CALL DISPLAY(2,44,'TARS',SC,SK,SK)
CALL DISPLAY(2,50,'SEA CO',SC,SK,SK)
CALL DISPLAY(2,59,'OTHER',SC,SK,SK)
CALL DISPLAY(2,67,'TOTAL',SC,SK,SK)
CALL DISPLAY(2,77,'MAX',SC,SK,SK)

```

```

SUMB = 0
CNT = 0

```

```

C-----
C The 10 and 20 loops load in the original allocation,
C for the selected ratings, into RATINGB and also sums up
C the total of the ratings. The numbers for RATINGB are
C integer for displaying, and therefore are rounded up to the
C nearest integer.

```

```

      DO 10 I=1,NUM_RATING
        RATINGB(I,PROG+1)=OUTYEAR(YEAR,1,RATING_INDEX(I),PROG+1)+0.5
10      SUMB = SUMB + RATINGB(I,PROG+1)

```

```

      DO 20 I=1,NUM_RATING
        DO 20 J=1,PROG
20      RATINGB(I,J) = OUTYEAR(YEAR,1,RATING_INDEX(I),J) + 0.5
C-----

```

```

      ROW = 2

```

```

C-----
C The 30 and 40 loops display the numbers for the Rating
C by program on the terminal. Also, the MAX is displayed.

```

```

      DO 40 I=1,NUM_RATING
        COLUMN = 1
        ROW = ROW + 2
        CALL DISPLAY(ROW,COLUMN,RATE_LABEL(RATING_INDEX(I)),SK,SK,SK)
        COLUMN = 2
        DO 30 K=1,PROG+1
          COLUMN = COLUMN + 8
          CALL DISPLAY_INTEGER(ROW,COLUMN,RATINGB(I,K),6,SK,SK,SK)
30      CONTINUE

```

```

        COLUMN = COLUMN + 8
        MA = MAX(YEAR,RATING_INDEX(I),8)

```

```

        IF (MA .LT. RATINGB(I,PROG+1)) THEN
          CALL DISPLAY_INTEGER(ROW,COLUMN,MA,6,BO,NE,SK)
        ELSE
          CALL DISPLAY_INTEGER(ROW,COLUMN,MA,6,SK,SK,SK)
        END IF

```

```

40      CONTINUE
C-----

```

```

      CALL DISPLAY(22,5,'Enter ROW and COLUMN to change or <Return>',
        SK,SK,SK)

```

```

CALL DISPLAY(22,52,'ROW _ COLUMN _',SK,SK,SK)

50  CONTINUE

CALL DISPLAY(22,67,' ',BO,SC,SK)

CALL ACCEPT_INTEGER(22,56,ROW,1,BO,SC,SK)
IF (ROW .LT. 1 .OR. ROW .GT. NUM_RATING) THEN
    GOTO 888
END IF

CALL ACCEPT_INTEGER(22,67,COLUMN,1,BO,SC,SK)
IF (COLUMN .LT. 1 .OR. COLUMN .GT. PROG+1) THEN
    GOTO 888
END IF

CNT = CNT + 1

J = ROW
K = COLUMN

C-----
C The 60 loop checks to see if the rating you have selected has been
C changed already. If no change yet then it adds it to the list of changed.
DO 60 L=1,NUM_C(YEAR-2)
    IF (RATING_INDEX(J) .EQ. CHANGED(YEAR-2,L)) THEN
        GOTO 70
    END IF
60  CONTINUE
C-----

NUM_C(YEAR-2) = NUM_C(YEAR-2) + 1
CHANGED(YEAR-2,NUM_C(YEAR-2)) = RATING_INDEX(J)

70  CONTINUE

ROW = (ROW * 2) + 2
COLUMN = (COLUMN * 8) + 2

CALL ACCEPT_INTEGER(ROW,COLUMN,RATINGB(J,K),6,BO,SC,SK)

C-----
C The if structure is checking to see if an individual program
C cell for a rating has been changed or if the total over all the
C Programs was changed. (PROG+1 is the total)
IF (K .EQ. PROG+1) THEN

    DIFFR = RATINGB(J,PROG+1) - OUTYEAR(YEAR,1,RATING_INDEX(J),PROG+1)
    OUTYEAR(YEAR,1,RATING_INDEX(J),PROG+1) = RATINGB(J,PROG+1)
    COLUMN = 2

C The 80 loop is reallocating the new total back over the programs for the
C rating then displaying the new numbers for the user.
DO 80 I=1,PROG
    OUTYEAR(YEAR,1,RATING_INDEX(J),I) =
*      OUTYEAR(YEAR,1,RATING_INDEX(J),I) +

```

```

      *      ((ALLOC(YEAR,RATING_INDEX(J),I)/
      *      ALLOC(YEAR,RATING_INDEX(J),PROG+1)) * DIFFR)
      RATINGB(J,I) = OUTYEAR(YEAR,1,RATING_INDEX(J),I) + .5
      COLUMN = COLUMN + 8
      CALL DISPLAY_INTEGER(ROW,COLUMN,RATINGB(J,I),6,BO,SC,SK)
80      CONTINUE

```

```

      ELSE

```

```

C If only a cell is changed then the total just has to be adjusted and
C displayed for the user.

```

```

      DIFFR = RATINGB(J,K) - OUTYEAR(YEAR,1,RATING_INDEX(J),K)
      OUTYEAR(YEAR,1,RATING_INDEX(J),PROG+1) =
      *      OUTYEAR(YEAR,1,RATING_INDEX(J),PROG+1) + DIFFR
      RATINGB(J,PROG+1)=OUTYEAR(YEAR,1,RATING_INDEX(J),PROG+1)+0.5
      OUTYEAR(YEAR,1,RATING_INDEX(J),K) = RATINGB(J,K)
      COLUMN = 66
      CALL DISPLAY_INTEGER(ROW,COLUMN,RATINGB(J,PROG+1),6,BO,SC,SK)

```

```

      END IF

```

```

C-----
C After a change check the MAX to flag the user if exceeded it.
      MA = MAX(YEAR,RATING_INDEX(J),8)

```

```

      IF (RATINGB(J,PROG+1) .GT. MA) THEN
          CALL DISPLAY_INTEGER(ROW,74,MA,6,BO,NE,SK)
      ELSE
          CALL DISPLAY_INTEGER(ROW,74,MA,6,SK,SK,SK)
      END IF

```

```

C-----

```

```

C-----
C The GOTO 50 sends the program back so the user can make another change.

```

```

      GOTO 50

```

```

C-----
C After the user has decided to exit changing rating the program
C sends the flow to 888.
888      CONTINUE

```

```

C If no changes were made then exit subroutine at 999.
      IF (CNT .EQ. 0) THEN
          GOTO 999
      END IF

```

```

C When there is a change for this outyear then flag it using DONE.
      DONE(YEAR-2) = 1

      SUMA = 0

```

```

C-----
C The 90 loop is calculating the sum of the ratings selected
C after the changes are made. Then the SUMB should = SUMA.
      DO 90 I=1,NUM_RATING

```

```

90      SUMA = SUMA + RATINGB(I,PROG+1)

      DIFF = SUMA - SUMB

C   The IF is checking to make sure the two sums are =.  If not the user is
C   asked if it is close enough.
      IF (DIFF .NE. 0) THEN

          CALL DISPLAY(23,5,'',SK,SK,SK)
          CALL DISPLAY(23,35,'',SK,SK,SK)
          CALL DISPLAY(23,65,'',SK,SK,SK)
          CALL DISPLAY(23,5,'The Total is',SK,SK,BE)
          CALL DISPLAY_INTEGER(23,18,SUMA,6,B0,SC,SK)
          CALL DISPLAY(23,25,'which is off by',SK,SK,SK)
          CALL DISPLAY_INTEGER(23,41,DIFF,6,B0,SC,SK)
          CALL DISPLAY(23,48,'Is this close enough (' ,SK,SK,SK)
          CALL DISPLAY(23,70,'Y',B0,SK,SK)
          CALL DISPLAY(23,71,' or',SK,SK,SK)
          CALL DISPLAY(23,75,'N',B0,SK,SK)
          CALL DISPLAY(23,76,'')?',SK,SK,SK)
          CALL ACCEPT(23,79,HOLD,B0,SC,SK)
          CALL DISPLAY(23,5,'',SK,SK,SK)
          CALL DISPLAY(23,35,'',SK,SK,SK)
          CALL DISPLAY(23,65,'',SK,SK,SK)

C   If the answer is not close enough GOTO 50 and make more changes.
      IF (HOLD .NE. 'Y' .AND. HOLD .NE. 'y') THEN
          GOTO 50
      END IF

      END IF

C-----
C-----
C   After the user is satisfied that the beginning and ending sums are
C   close enough or if they are equal, then the 100,110,120 loops are
C   used to recompute the totals for the LOS 1 of this outyear.
      DO 100 J=1,RATE
100      OUTYEAR(YEAR,1,J,PROG+1) = 0.0

      DO 110 K=1,PROG+1
110      OUTYEAR(YEAR,1,RATE+1,K) = 0.0

      DO 120 J=1,RATE
          DO 120 K=1,PROG
              OUTYEAR(YEAR,1,RATE+1,K) = OUTYEAR(YEAR,1,RATE+1,K) +
              OUTYEAR(YEAR,1,J,K)
              OUTYEAR(YEAR,1,J,PROG+1) = OUTYEAR(YEAR,1,J,PROG+1) +
              OUTYEAR(YEAR,1,J,K)
              OUTYEAR(YEAR,1,RATE+1,PROG+1) = OUTYEAR(YEAR,1,J,K) +
              OUTYEAR(YEAR,1,RATE+1,PROG+1)
          *
          *
          *
120      CONTINUE
C-----

999      CONTINUE

```

CALL CHANGE\_SCREEN(CL,SK,SK)

RETURN

END

```

C-----
C
C           S U B R O U T I N E   CHECK TO SEE IF NEW ALLOC
C
C-----
C
C      This subroutine asks the user if the allocation matrix being used in
C      this run should be saved on disk or not.
C
C      LOS_1_C(OUTY,RATE+1,PROG+1)      This matrix is passed into this sub.
C                                         and it contains the matrix to be used
C                                         in this run.
C
C      NEW_LOS1(OUTY,RATE+1,PROG+1)      This matrix is filled with LOS_1_C and
C                                         then is used to work with in creating
C                                         the new allocation.
C
C      NEW_ALLOC(20,RATE+1,PROG+1)      This matrix will store the allocation
C                                         percents and is written to disk for
C                                         use in a later run.
C-----

```

```

SUBROUTINE CHECK_TO_SEE_IF_SAVE_NEW_ALLOC(LOS_1_C)

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

REAL LOS_1_C(OUTY,RATE+1,PROG+1), NEW_ALLOC(20,RATE+1,PROG+1),
•    NEW_LOS1(OUTY,RATE+1,PROG+1)

```

```

CHARACTER ANS

```

```

CHARACTER*50 FILEN
CHARACTER*25 NAME

```

```

CALL CHANGE_SCREEN(CL,SK,SK)

```

```

C-----
C  The first thing is to ask the user if this allocation should be
C  saved.  If not then exit subroutine.
C      CALL DISPLAY(10,10,'Would you like to save this allocation (',
•          SK,SK,SK)
C      CALL DISPLAY(10,50,'Y',BO,SK,SK)
C      CALL DISPLAY(10,51,'es or',SK,SK,SK)
C      CALL DISPLAY(10,57,'N',BO,SK,SK)
C      CALL DISPLAY(10,58,'o)?',SK,SK,SK)
5      CONTINUE
      CALL ACCEPT(10,62,ANS,BO,SC,SK)
      IF (ANS .NE. 'N' .AND. ANS .NE. 'Y' .AND. ANS .NE. 'n' .AND.
•          ANS .NE. 'y') THEN
          GOTO 5
      END IF
      IF (ANS .EQ. 'N' .OR. ANS .EQ. 'n') THEN
          GOTO 999

```

```

        END IF

        CALL DISPLAY(14,5,'Enter a name to save this allocation under',
                     SK,SK,SK)
10      CONTINUE
        NAME = '
        CALL ACCEPT(14,48,NAME,B0,NE,SK)

        IF (NAME .EQ. '          ') THEN
            GOTO 10
        END IF

        FILEN = 'ZZZ_'//NAME

        OPEN(UNIT=60,STATUS='NEW',FILE=FILEN,FORM='UNFORMATTED',ERR=10)

C-----
C  The 20 loop sets the NEW_LOS1 matrix equal to the passed in LOS_1_C matrix,
C  so we can work with NEW_LOS1.
        DO 20 K=1,PROG+1
            DO 20 J=1,RATE+1
                DO 20 I=1,OUTY
20          NEW_LOS1(I,J,K) = LOS_1_C(I,J,K)
C-----

C-----
C  The 30 and 40 loops subtract out the Sea College new recruits
C  because they do not work like the rest of the programs.
        DO 40 I=1,OUTY
            NEW_LOS1(I,RATE+1,PROG+1) =
            *   NEW_LOS1(I,RATE+1,PROG+1) - NEW_LOS1(I,RATE+1,6)
            NEW_LOS1(I,RATE+1,6) = 0.0

            DO 30 J=1,RATE
                NEW_LOS1(I,J,PROG+1) = NEW_LOS1(I,J,PROG+1) -
            *   NEW_LOS1(I,J,6)
                NEW_LOS1(I,J,6) = 0.0
30          CONTINUE
40          CONTINUE
C-----

C-----
C  The 50 loop actually creates the new allocation percents to be
C  written to disk for uses later.
        DO 50 K=1,PROG+1
            DO 50 J=1,RATE+1
                DO 50 I=1,OUTY
                    IF (NEW_LOS1(I,RATE+1,PROG+1) .GT. 0) THEN
                        NEW_ALLOC(I,J,K) = NEW_LOS1(I,J,K) /
                        NEW_LOS1(I,RATE+1,PROG+1)
                    ELSE
                        NEW_ALLOC(I,J,K) = 0.0
                    END IF
50          CONTINUE

```

```

C-----
C-----
C The 60 loop makes every year passed OUTY = to the allocation at
C The OUTY dimension.
      DO 60 K=1,PROG+1
          DO 60 J=1,RATE+1
              DO 60 I=OUTY+1,20
60          NEW_ALLOC(I,J,K) = NEW_ALLOC(OUTY,J,K)
C-----

      WRITE(60) NEW_ALLOC
      CLOSE(UNIT=60)

999    CONTINUE

      RETURN
      END

```



---

S U B R O U T I N E   T R A N S I T I O N

---

This is the main subroutine of this program. This subroutine transitions the history and new recruits that the user has inputed.

I, J, K, L, M, N                      These variables are used as indexes into arrays.

SUB\_LOSS(PROG)                      This array is used to show how the Programs need to be collapsed for the LOSS matrix.

ACCESS(OUTY)                      This array is passed to this subroutine and it contains the number of recruits that the user input for each outyear.

TRANS(LOS-2,RATE,PROG)              This array is used to store the continuation rates from one SEPT. to the next. Is capped at 1.0. It is passed into the subroutine.

N\_D(LOS-2,RATE,PROG)              This array is to show what percent is to be added back. This array is 0.0 except where TRANS is = 1.0. This array is also passed into the subroutine.

OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1)      This array is passed into this subroutine and it has two years of History and LOS 1 filled out for each outyear. After this subroutine this array will be filled with the transitioned history.

PRIOR\_LOS(LOS,RATE)              This array will be read into this subroutine and it contains the percents to show how the newblood Prior Service break out into LOS and RATING.

ADDER(LOS,RATE,PROG)              This array is recreated for each outyear of the simulation. It will contain numbers that show what to add back to each Rating and Program combination in each LOS.

LOSSES(OUTY,RATE,PROG-3)              This is used to compute the losses for each outyear. This array will be written out to be used by the reserve simulation.

REALNUM                      Is used for computing purposes.

```

C      SEA_C(RATE,RATE)                This matrix shows how and where the
C                                         user inputed SEA COLLEGE recruits will
C                                         end up in LOS 3.
C-----

```

```

SUBROUTINE TRANSITION(ACCESS,OUTYEAR,LOSSES)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

INTEGER I, J, K, L, M, N, SUB_LOSS(PROG), ACCESS(OUTY)

```

```

REAL TRANS(LOS-2,RATE,PROG,3), LAT(LOS-2,PROG),

```

```

*      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1), PRIOR_LOS(LOS,RATE),
*      ADDER(LOS-2,PROG), LOSSES(OUTY,RATE,PROG-3), REALNUM,
*      SEA_C(RATE,RATE)

```

```

DATA SUB_LOSS/1,2,1,1,1,3,4/

```

```

CALL CHANGE_SCREEN(CL,SK,SK)

```

```

CALL DISPLAY(6,15,'Please wait while simulation takes place',
*              BO,NE,SK)

```

```

CALL DISPLAY(10,15,'Beginning to transition the history',BO,SK,SK)

```

```

C--- Initializing

```

```

DO 10 K=1,PROG-3

```

```

    DO 10 J=1,RATE

```

```

        DO 10 I=1,OUTY

```

```

10          LOSSES(I,J,K) = 0.0

```

```

C-----

```

```

*      OPEN(UNIT=59,STATUS='OLD',FORM='UNFORMATTED',READONLY,
*          FILE='CNA2:[CORLISSG.FORCE.EMR]MATCH8586_PCTS.DAT')
*      READ(59) TRANS
*      CLOSE(59)

```

```

*      OPEN(UNIT=59,STATUS='OLD',READONLY,FORM='UNFORMATTED',
*          FILE='CNA2:[CORLISSG.FORCE.EMR]MATCH8586_LAT_PCTS.DAT')
*      READ(59) LAT
*      CLOSE(59)

```

```

*      OPEN (UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
*          FILE='CNA2:[CORLISSG.FORCE.DAT]PRIOR_LOS.DAT')
*      READ(59) PRIOR_LOS
*      CLOSE(UNIT=59)

```

```

*      OPEN (UNIT=59,READONLY,STATUS='OLD',FORM='UNFORMATTED',
*          FILE='CNA2:[CORLISSG.FORCE.DAT]SEA_C.DAT')
*      READ(59) SEA_C
*      CLOSE(UNIT=59)

```

```

CALL CREATE_ADDER(2,OUTYEAR,LAT,ADDER)

```

```

C-----
C The 90 and 40 loops are the main loops of
C this simulation. This is where the history
C is transitioned into to future.
      DO 90 I=3,OUTY+2
          DO 40 L=1,PROG
              DO 40 K=1,RATE

C First thing is to create LOS 2 at 91.6% survival from last years LOS 1.
C 0.903 was created by taking the 85 LOS 1 -> LOS 2
C continuation rates for all programs. 0.908 IS ATTRITION FROM LOS 0 -> LOS 1.

      OUTYEAR(I,2,K,L) = OUTYEAR(I-1,1,K,L) * 0.903 * 0.908

C-----
C Calculating LOS 3 for each of the outyears. It is nessary to break out
C LOS 3 because the first transition from 1 --> 3 is taken from Two years
C ago and the rest of the LOS's are computed from 1 year ago.
C
      OUTYEAR(I,3,K,L) = OUTYEAR(I-2,1,K,L) * TRANS(1,K,L,1)

      LOSSES(I-2,K,SUB_LOSS(L)) = LOSSES(I-2,K,SUB_LOSS(L)) +
      * (OUTYEAR(I-2,1,K,L) * TRANS(1,K,L,2))

      OUTYEAR(I,3,K,L) = OUTYEAR(I,3,K,L) +
      * (ADDER(1,L) * TRANS(1,K,L,3))

      OUTYEAR(I,1,K,L) = OUTYEAR(I,1,K,L) * 0.908

C-----
C-----
C The 20 loop calculates LOS 4-9. LOS's 4 to 9 are unique because all the
C programs are kept in the transition rates.
C
      DO 20 J=4,9

      OUTYEAR(I,J,K,L) = OUTYEAR(I-1,J-1,K,L) * TRANS(J-2,K,L,1)

      LOSSES(I-2,K,SUB_LOSS(L)) = LOSSES(I-2,K,SUB_LOSS(L)) +
      * (OUTYEAR(I-1,J-1,K,L) * TRANS(J-2,K,L,2))

      OUTYEAR(I,J,K,L) = OUTYEAR(I,J,K,L) +
      * (ADDER(J-2,L) * TRANS(J-2,K,L,3))

20      CONTINUE

C-----
C-----
C The 30 loop calculates OUTYEAR for LOS10-LOS31. This loop is nessary
C because after LOS 9 all programs transition to program 1, and then
C only Program 1 is used.
      DO 30 J=10,LOS

      OUTYEAR(I,J,K,1) = OUTYEAR(I,J,K,1) +
      * (OUTYEAR(I-1,J-1,K,L) * TRANS(J-2,K,L,1))

```

```

        LOSSES(I-2,K,SUB_LOSS(L)) = LOSSES(I-2,K,SUB_LOSS(L)) +
        *      (OUTYEAR(I-1,J-1,K,L) * TRANS(J-2,K,L,2))

        OUTYEAR(I,J,K,1) = OUTYEAR(I,J,K,1) +
        *      (ADDER(J-2,L) * TRANS(J-2,K,L,3))

30          CONTINUE

40          CONTINUE

C-----
C  The 50 loop is nessary to take the SEA COLLEGE recruits and move them to the
C  appropriate ratings in two years. This loop does the 1 --> 3 transition for
C  Sea College using the SEA_C matrix which is read into the program.
        DO 50 J=1,RATE
            DO 50 K=1,RATE
50          OUTYEAR(I,3,K,6) = OUTYEAR(I,3,K,6) +
        *      (OUTYEAR(I-2,1,J,6) * SEA_C(J,K))
C-----
C-----
C  The next set of loops are used to take the Newblood Prior Service people
C  and redistribuite them to the proper LOS, and Rating. The Newbloods are
C  sitting at LOS 1 before this loop and the PRIOR_LOS matrix is read into the
C  program to show where the individuals go.

        OUTYEAR(I,1,RATE+1,PROG+1) = OUTYEAR(I,1,RATE+1,PROG+1) -
        *      OUTYEAR(I,1,RATE+1,4)

        DO 80 K=1,RATE
            REALNUM = OUTYEAR(I,1,K,4)
            OUTYEAR(I,1,K,4) = 0.0

            DO 60 J=1,9
60          OUTYEAR(I,J,K,4) = OUTYEAR(I,J,K,4) +
        *      (REALNUM * PRIOR_LOS(J,K))
C-----
C  After LOS 9 only Program 1 exists.
            DO 70 J=10,LOS
70          OUTYEAR(I,J,K,1) = OUTYEAR(I,J,K,1) +
        *      (REALNUM * PRIOR_LOS(J,K))

80          CONTINUE

C-----
C  Call CREATE_ADDER to get the new additions for the next outyear
        IF (I .NE. OUTY+2) THEN
            CALL CREATE_ADDER(I,OUTYEAR,LAT,ADDER)
        END IF
90      CONTINUE

        CALL DISPLAY(11,15,'Finished creating the outyear inventories',
        *      BO,SK,SK)

        RETURN
    END

```

```

C-----
C
C           S U B R O U T I N E   C R E A T E   A D D E R
C-----
C
C      This subroutine is called by TRANSITION and it creates an array to show
C      how many individuals to add back for the next outyear.
C
C      I, J, K           These variables are used as indexes
C                        into arrays.
C
C      YEAR              Is used to determine which year is
C                        being worked with. It is passed into
C                        this subroutine.
C
C      ACCESS(OUTY)      It is passed to this subroutine and it
C                        shows what the user input for new
C                        recruits for each outyear.
C
C      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1) Is used to calculate the rating
C                        and LOS 1 bases needed for N_D.
C
C      N_D(LOS-2,RATE,PROG) This array is to show what percent is
C                        to be aded back. This array is 0.0
C                        except where TRANS is = 1.0. This
C                        array is passed into this subroutine.
C
C      BIG_T             Will store the total accessed at YEAR-2.
C
C      ADDER(LOS,RATE,PROG) It will contain numbers that show what
C                        to add back to each Rating and Program
C                        combination in each LOS. Is computed
C                        using N_D.
C
C      TOT(RATE)         This is used to store the total number
C                        in each rating for YEAR-1. This is
C                        needed by N_D for LOS's 4-los.
C-----

```

```

SUBROUTINE CREATE_ADDER(YEAR,OUTYEAR,LAT,ADDER)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

INTEGER I, J, K, L, YEAR

```

```

REAL OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1),

```

```

*      ADDER(LOS-2,PROG), INV(LOS-2,PROG), LAT(LOS-2,PROG)

```

```

C--- Initializing

```

```

DO 10 I=1,LOS-2

```

```

    DO 10 J=1,PROG

```

```

10      INV(I,J) = 0.0

```

```

DO 15 K=1,PROG
    DO 15 J=1,RATE
15      INV(1,K) = INV(1,K) + OUTYEAR(YEAR,1,J,K)

DO 20 K=1,PROG
    DO 20 J=1,RATE
        DO 20 I=3,LOS-1
20      INV(I-1,K) = INV(I-1,K) + OUTYEAR(YEAR,I,J,K)
C-----

DO 30 K=1,PROG
    DO 30 I=1,LOS-2
        IF ((I .LT. 10) .OR. (K .EQ. 1)) THEN
            ADDER(I,K) = INV(I,K) * LAT(I,K)
        END IF
30      CONTINUE
C-----

RETURN
END

```

```

C-----
C
C          S U B R O U T I N E   A D D   P A Y G R A D E
C-----
C
C      This subroutine will take the inventory created in the simulation
C      and break it out by PAYGRADE so it can be compared to requirements.
C
C      I, J, K, L, M          These are used as indexes into
C                             arrays.
C
C      OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1) This array will be passed into the
C                                       subroutine and it contains the
C                                       simulated outyear inventories.
C
C      PAYGRADE(LOS,RATE,PROG,PAYG) This array will be read into the
C                                   subroutine and it contains the percents
C                                   to show how paygrade is to be broken
C                                   out of the inventory. Each outyear
C                                   of the inventory uses the same breakout.
C
C      SUM(OUTY,RATE,PAYG) This array will be used to store the
C                           new inventory by RATE, PAYGRADE and
C                           this is passed out of the subroutine
C                           to be used to compare to requirements.
C-----

```

```

SUBROUTINE ADD_PAYGRADE(OUTYEAR,SUM)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

```

```

INTEGER I, J, K, L, M

```

```

REAL OUTYEAR(OUTY+2,LOS,RATE+1,PROG+1),
•    PAYGRADE(LOS,RATE,PROG,PAYG), SUM(OUTY,RATE,PAYG)

```

```

CALL DISPLAY(13,15,'Beginning to add paygrade to outyear inventories',
•    SK,SK,SK)

```

```

OPEN (UNIT=56,READONLY,STATUS='OLD',FORM='UNFORMATTED',
•    FILE='CNA2:[CORLISSG.FORCE.DAT]PAYGRADE.DAT')
READ(56) PAYGRADE
CLOSE(UNIT=56)

```

```

DO 10 M=1,PAYG
    DO 10 L=1,PROG
        DO 10 K=1,RATE
            DO 10 J=1,LOS
                DO 10 I=3,OUTY+2

```

```

10    SUM(I-2,K,M) = SUM(I-2,K,M) + (OUTYEAR(I,J,K,L) * PAYGRADE(J,K,L,M))

```

```

CALL DISPLAY(14,15,'Finished adding paygrade to outyear inventories',

```

• SK, SK, SK)

RETURN  
END



---

S U B R O U T I N E   W R I T E I T

---

This subroutine is used to write out to the screen and to a disk file the results of the simulation. The simulated results will be compared to the NAVY requirements.

RATING\_INDEX(RATE)                      This array is used to store which ratings the user has selected to view on the terminal.

NUM\_RATING                                This is a count of how many ratings the user selected to view.

I, J, K, L                                These variables are used as indexes into arrays.

OUT                                        This stores the year that the user selected for viewing on the terminal.

NUM\_C(OUTY)                              This is passed into the subroutine and it contains the counts for the number of ratings changed for each outyear.

CHANGED(OUTY,RATE)                      This array stores the ratings that were changed for each outyear.

CNTR                                      This is used to compute how many positions to zero out in certain arrays.

CNT                                        This is used to keep a count of where a page break is needed in the output.

REQ(20,RATE,PAYG)                      This array is read into the program in this subroutine and it contains the NAVY requirements.

SUM(OUTY,RATE,PAYG)                      This array is passed into this subroutine and it contains the simulated results for each outyear by Rate and Paygrade.

S(OUTY+1,RATE+1,PAYG+1)                This array is the same as SUM except that Totals have been added.

SR(OUTY+1,RATE+1,PAYG+1)               This array is the same as REQ except that Totals have been added.

DIFF(OUTY+1,RATE+1,PAYG+1)             This array is computed by subtracting SR from S in each cell.

RATIO(OUTY+1,RATE+1,PAYG+1)            This array is computed by dividing DIFF by SR at every cell where SR > 0.0

```

C
C      LOS_1(OUTY,RATE+1,PROG+1)      This array is passed into this
C                                     subroutine and it contains the original
C                                     LOS 1 allocation of new recruits for
C                                     each of the outyears.
C
C      LOS_1_C(OUTY,RATE+1,PROG+1)    This array is passed into this
C                                     subroutine and it contains the original
C                                     LOS 1 allocation except where the user
C                                     has changed a Rating or Program mix.
C
C      YEAR_SEL*2                      This is passed into the subroutine
C                                     and it contains the year for the
C                                     transitions that the user selected.
C-----

```

```

      SUBROUTINE WRITEIT(SUM,NUM_C,CHANGED,YR_SEL,LOS_1,LOS_1_C,LOSSES)

      IMPLICIT NONE

      INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
      INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

      INTEGER RATING_INDEX(RATE), NUM_RATING

      INTEGER I, J, K, L, OUT, REQ(20,RATE,PAYG), NUM_C(OUTY),
      *      CHANGED(OUTY,RATE), CNT, CNTR

      REAL SUM(OUTY,RATE,PAYG), DIFF(OUTY+1,RATE+1,PAYG+1),
      *      RATIO(OUTY+1,RATE+1,PAYG+1),S(OUTY+1,RATE+1,PAYG+1),
      *      SR(OUTY+1,RATE+1,PAYG+1), LOSSES(OUTY,RATE,PROG-3),
      *      LOS_1(OUTY,RATE+1,PROG+1), LOS_1_C(OUTY,RATE+1,PROG+1)

      PARAMETER(CNTR=(OUTY+1)*(RATE+1)*(PAYG+1))

      CHARACTER HOLD, ANS, FLAG
      CHARACTER*2 YR_SEL
      CHARACTER*8 RATE_LABEL

      CHARACTER*25 NAME
      CHARACTER*29 FILEN

      DATA S/CNTR*0.0/,SR/CNTR*0.0/,DIFF/CNTR*0.0/,RATIO/CNTR*0.0/

      CALL DISPLAY(16,15,'Beginning to write results to disk',B0,SK,SK)

      OPEN(UNIT=50,FILE='CNA2:[CORLISSG.FORCE.DAT]REQ.DAT',
      *      FORM='UNFORMATTED',READONLY,STATUS='OLD')
      READ(50) REQ
      CLOSE(UNIT=50)

```

```

C-----
C      The 10 loops will sum up everything needed

      DO 10 K=1,PAYG
        DO 10 J=1,RATE

```

```

DO 10 I=1,OUTY
S(I,J,K) = S(I,J,K) + SUM(I,J,K)
S(I,RATE+1,K) = S(I,RATE+1,K) + SUM(I,J,K)
S(I,J,PAYG+1) = S(I,J,PAYG+1) + SUM(I,J,K)
S(OUTY+1,J,K) = S(OUTY+1,J,K) + SUM(I,J,K)
S(OUTY+1,RATE+1,K) = S(OUTY+1,RATE+1,K) + SUM(I,J,K)
S(I,RATE+1,PAYG+1) = S(I,RATE+1,PAYG+1) + SUM(I,J,K)
S(OUTY+1,J,PAYG+1) = S(OUTY+1,J,PAYG+1) + SUM(I,J,K)
S(OUTY+1,RATE+1,PAYG+1) = S(OUTY+1,RATE+1,PAYG+1) + SUM(I,J,K)

SR(I,J,K) = SR(I,J,K) + REQ(I,J,K)
SR(I,RATE+1,K) = SR(I,RATE+1,K) + REQ(I,J,K)
SR(I,J,PAYG+1) = SR(I,J,PAYG+1) + REQ(I,J,K)
SR(OUTY+1,J,K) = SR(OUTY+1,J,K) + REQ(I,J,K)
SR(OUTY+1,RATE+1,K) = SR(OUTY+1,RATE+1,K) + REQ(I,J,K)
SR(I,RATE+1,PAYG+1) = SR(I,RATE+1,PAYG+1) + REQ(I,J,K)
SR(OUTY+1,J,PAYG+1) = SR(OUTY+1,J,PAYG+1) + REQ(I,J,K)
10 SR(OUTY+1,RATE+1,PAYG+1) = SR(OUTY+1,RATE+1,PAYG+1) + REQ(I,J,K)

```

```

C-----
C      The next section is for creating a disk file that shows the
C      comparison of simulated results to requirements

```

```

OPEN (UNIT=66,STATUS='NEW',FILE='ACTIVE.DAT')

```

```

DO 20 K=1,PAYG+1
DO 20 J=1,RATE+1
DO 20 I=1,OUTY+1

DIFF(I,J,K) = S(I,J,K) - SR(I,J,K)

IF (SR(I,J,K) .EQ. 0) THEN
RATIO(I,J,K) = 0.0
ELSE
RATIO(I,J,K) = DIFF(I,J,K) / SR(I,J,K)
END IF

```

```

20 CONTINUE

```

```

DO 80 I=1,OUTY
WRITE(66,200)
WRITE(66,180)
WRITE(66,190) I + (CFY - 1)
WRITE(66,180)
WRITE(66,180)
WRITE(66,250)
WRITE(66,260)
WRITE(66,220)
WRITE(66,260)

DO 40 J=1,RATE+1

FLAG = ' '

DO 30 L=1,NUM_C(I)
IF (CHANGED(I,L) .EQ. J) THEN

```

```

                                FLAG = '*'
                                END IF
30      CONTINUE

                                WRITE(66,230) RATE_LABEL(J),
                                *      (LOS_1(I,J,K),K=1,PROG+1),
                                *      (LOS_1_C(I,J,K),K=1,PROG+1),FLAG

40      CONTINUE

                                CNT = 60

                                DO 60 J=1,RATE

                                    CNT = CNT + 1

                                    IF (CNT .GT. 52) THEN
                                        WRITE(66,200)
                                        WRITE(66,180)
                                        WRITE(66,270)
                                        WRITE(66,180)
                                        CNT = 1
                                    END IF

C      SKIPPING PAYGRADE WRITEOUT FOR EACH RATING
C
C      IF (CNT .NE. 1) THEN
C          WRITE(66,180)
C      END IF
C
C      DO 50 K=1,PAYG
C          WRITE(66,170) RATE_LABEL(J), S(I,J,K),
C      *      SR(I,J,K), DIFF(I,J,K), RATIO(I,J,K)
C50      CONTINUE
C
C      WRITE(66,180)

                                WRITE(66,170) RATE_LABEL(J), S(I,J,PAYG+1),
                                *      SR(I,J;PAYG+1), DIFF(I,J,PAYG+1),
                                *      RATIO(I,J,PAYG+1)

60      CONTINUE

                                WRITE(66,180)

                                DO 70 K=1,PAYG
                                    WRITE(66,170) 'OUTY PG ',S(I,RATE+1,K),SR(I,RATE+1,K),
                                    *      DIFF(I,RATE+1,K), RATIO(I,RATE+1,K)
70      CONTINUE

                                WRITE(66,180)
                                WRITE(66,170) 'TOTAL OY', S(I,RATE+1,PAYG+1),
                                *      SR(I,RATE+1,PAYG+1), DIFF(I,RATE+1,PAYG+1),
                                *      RATIO(I,RATE+1,PAYG+1)

80      CONTINUE

```

```

        WRITE(66,180)
        DO 90 K=1,PAYG
            WRITE(66,170) 'TOT PG ', S(7,RATE+1,K), SR(7,RATE+1,K),
                *           DIFF(7,RATE+1,K), RATIO(7,RATE+1,K)
90      CONTINUE

        WRITE(66,180)
        WRITE(66,170) 'TOT TOT ', S(OUTY+1,RATE+1,PAYG+1),
            *           SR(OUTY+1,RATE+1,PAYG+1), DIFF(OUTY+1,RATE+1,PAYG+1),
            *           RATIO(OUTY+1,RATE+1,PAYG+1)

        CLOSE (UNIT=66)

        CALL DISPLAY(17,15,'Finished writing results to disk',BO,SK,SK)

C-----
C      Writing out the sums of the simulated results to disk for use in the
C      Compare program.

        CALL DISPLAY(19,5,'Enter a filename for storing the results',
            *           SK,SK,SK)
        CALL DISPLAY(19,46,'(For use later in the Compare run)',SK,SK,SK)
100     CONTINUE
        NAME = '
        CALL ACCEPT(21,20,NAME,BO,NE,SK)

        IF (NAME.EQ. '
            *           ') THEN
            GOTO 100
        END IF

        FILEN = 'ACT_'//NAME

        OPEN (UNIT=67,STATUS='NEW',FILE=FILEN,FORM='UNFORMATTED',ERR=100)
        WRITE(67) S
        CLOSE (UNIT=67)

        FILEN = 'LOS_'//NAME

C-----
C      Write out LOSSES to be used by RESERVE simulation.
        OPEN(UNIT=67,STATUS='NEW',FILE=FILEN,FORM='UNFORMATTED')
        WRITE(67) LOSSES
        CLOSE(UNIT=67)

170     FORMAT(4X,A8,4F17.3)
180     FORMAT(1X)
190     FORMAT(55X,'Y E A R ',I4)
200     FORMAT('1')
210     FORMAT(40X,'***** No Rating Mixes were changed *****')
220     FORMAT(11X,' 4 YO ', ' Act Ma', ' 5&6Y0', ' Prior', ' Tars',
            *           ' Sea C', ' Other', ' Total',3X,'|',2X,
            *           ' 4 YO ', ' Act Ma', ' 5&6Y0', ' Prior', ' Tars',
            *           ' Sea C', ' Other', ' Total')

230     FORMAT(1X,A8,2X,8F7.0,3X,'|',2X,8F7.0,1X,A1)
240     FORMAT(40X,'(Transition Rates for Fiscal Year ',A2,' were used)')

```

```

250  FORMAT(23X,'O R I G I N A L',32X,'|',25X,'C H A N G E D')
260  FORMAT(70X,'|')
270  FORMAT(12X,'          Simulated', '          Requirements',
*      '          Difference', '          Percent Diff')

999  CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      RETURN
      END

```

ANNEX D-3

LISTING OF RESERVE (SELRES) SIMULATION PROGRAM  
(VAX-11 FORTRAN)

---

E N L I S T M E N T   P R O G R A M S   S I M U L A T I O N :  
R E S E R V E   ( S E L R E S )   P R O G R A M

---

This program is set up to accept inputs via the terminal for Reserve recruits in the SAMS and OTHER programs. This program also reads in the Active simulation losses for use in the Reserve simulation.

SAMS(OUTY)                      This array is used to store the numbers input by the user for each outyear. The numbers are for new SAM recruits.

OTHER(OUTY)                    This array contains The user inputs for new recruits into the OTHER program.

OUTYEAR(OUTY+1,LOS,RATE,PROG)   This array will store the most recent year of history in the first outyear dimension. The other outyear dimensions will be filled by simulating the history forward into the future. The LOS 1 will be filled by user inputs and Active simulation losses.

LOSSES(OUTYEAR,RATE,PROG-2)   This array will be read into the program and it contains the Active simulation losses for each outyear.

SUM(OUTY,RATE,PAYG)            This array is created after the OUTYEAR array has been created. It will be the same as OUTYEAR except that it is collapsed over program and paygrade is added. It is used for comparing to requirements.

ACTFILE                        This stores the filename of the Active loss file selected. Is used for displaying to the user.

---

IMPLICIT NONE

---

The INCLUDE for Screen\_Parameters.for contains the variables used for the screen handling subroutines. They are:

BE = 'BE'    used to sound the bell  
BL = 'BL'    used to make the screen blink  
BO = 'BO'    used to make the screen bold  
CL = 'CL'    used for clearing the screen  
NE = 'NE'    used to make the screen negative  
NO = 'NO'    used to set screen back to normal  
SC = 'SC'    used to score (underline) data on screen



```

C      SK = ' ' used to skip an option.
C-----
C      INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

C-----
C  The INCLUDE for Parameters_Res.for contains the variables
C  to show how the Arrays should be dimensioned.  They are:
C
C      LOS = 8      There are 8 LOS's in the Reserves
C      RATE = 69     There are 69 Ratings
C      PROG = 6      There are 6 programs in the Reserves
C      PAYG = 9      There are 9 paygrades
C      OUTY = 10     There are 10 outyears for this simulation
C-----
C      INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS_RES.FOR'

      INTEGER SAMS(OUTY), OTHER(OUTY)

      REAL OUTYEAR(OUTY+1,LOS,RATE,PROG), SUM(OUTY,RATE,PAYG),
*      LOSSES(OUTY,RATE,PROG-2), ALLOC_SAMS(RATE)

      CHARACTER HOLD, ANS
      CHARACTER*50 ACTFILE

      OPEN (UNIT=6,STATUS='OLD',RECL=500)

```

C-----  
C  
C  
C  
C-----

M A I N L O G I C

CALL CHANGE\_SCREEN(CL,SK,SK)  
CALL DISPLAY(10,15,'R E S E R V E F O R C E S I M U L A T I O N',  
BO,SK,SK)  
CALL DISPLAY(22,25,'HIT RETURN TO START RUN',SK,SK,SK)  
CALL ACCEPT(22,49,HOLD,SK,SK,SK)

CALL GET\_LOSSES(LOSSES,ACTFILE)

CALL ACCEPT\_OUTYEAR\_RECRUITS(OUTYEAR,SAMS,OTHER,LOSSES)

CALL CHANGE\_M'YES(OUTYEAR)

CALL TRANSITION(OUTYEAR)

CALL ADD\_PAYGRADE(OUTYEAR,SUM)

CALL WRITE\_RESULTS(SUM,SAMS,OTHER,LOSSES,ACTFILE)

CALL CHANGE\_SCREEN(CL,SK,SK)  
CALL DISPLAY(10,20,'FINISH OF RESERVE FORCE SIMULATION',BO,SK,SK)  
CALL DISPLAY(23,25,'HIT RETURN TO FINISH',SK,SK,SK)  
CALL ACCEPT(23,49,HOLD,SK,SK,SK)  
CALL CHANGE\_SCREEN(CL,SK,SK)

STOP  
END

```

C-----
C
C           S U B R O U T I N E   G E T   L O S S E S
C
C-----
C
C      This subroutine reads in a user selected Active loss file and passes
C      the loss file along with the filename out of the routine.
C
C      LOSSES(OUTY,RATE,PROG-2)      This file contains the Active losses
C                                     for a specific run that the user has
C                                     selected to use.
C
C      FILEN                        Has the filename of the user selected
C                                     loss file that is to be opened.
C
C      ACTFILE                      This is the filename to display to
C                                     the user.
C-----

```

```

SUBROUTINE GET_LOSSES(LOSSES,ACTFILE)

```

```

  IMPLICIT NONE

```

```

  INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.RES.FOR'

```

```

  INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

  REAL LOSSES(OUTY,RATE,PROG-2)

```

```

  CHARACTER HOLD

```

```

  CHARACTER*50 FILEN, ACTFILE

```

```

  CALL FIND_ACTIVE_LOSS_FILES(FILEN,ACTFILE)

```

```

  OPEN(UNIT=50,STATUS='OLD',FILE=FILEN,FORM='UNFORMATTED',
    •      READONLY)

```

```

  READ(50) LOSSES

```

```

  CLOSE(UNIT=50)

```

```

  RETURN

```

```

  END

```

```

C-----
C
C      SUBROUTINE FIND_ACTIVE_LOSS_FILES
C-----
C      This subroutine is used to allow the user to select any previously
C      created Active loss matrix. The subroutine searches the
C      users directory for loss files and displays to the terminal the
C      choices. If no choices are available, execution of this program
C      is terminated.
C
C      FILEN                This is passed back to the main
C                           program and it contains the filename
C                           of the users choice or 'FLAG'.
C
C      NEWFILE              This is used in the library call and
C                           it contains the next file found in the
C                           directory. When no more files it
C                           passes back the default filename.
C
C      DEFAULT, RELATED, FILENAME  These are used by the library and are
C                           used to locate a certain set of files.
C
C      STORE(100)           This is used to store the names of all
C                           the user created allocations found
C                           in the directory.
C
C      DIS(100)             This stores the names of the files to
C                           be displayed to the user on the screen.
C
C      CHECK                Is used to check and see when there are
C                           no more files.
C
C      CONTEXT              Is used by the library subroutine as an
C                           address pointer and must be set to 0
C                           at start.
C
C      ROW, ROW1            Used to calculate which row of the
C                           screen to display on.
C
C      BEGIN, END           Used to store the begin and end of
C                           strings that are being searched for in
C                           other strings.
C
C      NUMFILES             Is a counter used to keep count of the
C                           of allocations found in the directory.
C
C      ANS                  The answer to which allocation the user
C                           selected to use.
C
C      I, D, FF             Used as integers for loops and
C                           calculations.
C-----

```

```

SUBROUTINE FIND_ACTIVE_LOSS_FILES(FILEN,ACTFILE)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2 [CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

INTEGER CONTEXT, ROW, ROW1, BEGIN, END, NUMFILES, ANS, I, DIFF

CHARACTER HOLD

CHARACTER*50 FILENAME, NEWFILE, DEFAULT, RELATED, FILEN,
•          STORE(100), DIS(100), CHECK, ACTFILE

CALL CHANGE_SCREEN(CL,SK,SK)

FILENAME = 'LOS_.DAT'

DEFAULT = FILENAME
RELATED = FILENAME

CONTEXT = 0
NUMFILES = 1

CHECK = '

10  CONTINUE

CALL LIB$FIND_FILE(FILENAME,NEWFILE,CONTEXT,DEFAULT,RELATED)

BEGIN = INDEX(NEWFILE,']LOS_')
END = INDEX(NEWFILE,'.DAT')
BEGIN = BEGIN + 1
END = END + 3
DIFF = (END - BEGIN) + 1

CHECK(1:DIFF) = NEWFILE(BEGIN:END)
DIS(NUMFILES) = CHECK(5:DIFF-4)

IF (CHECK .NE. FILENAME) THEN
  IF (NEWFILE .NE. STORE(NUMFILES)) THEN
    STORE(NUMFILES) = NEWFILE
    NUMFILES = NUMFILES + 1
  END IF
  CHECK = '
  GOTO 10
END IF

NUMFILES = NUMFILES - 1

IF (NUMFILES .EQ. 0) THEN
  CALL CHANGE_SCREEN(CL,SK,SK)
  CALL DISPLAY(12,10,'          Cannot run Reserve simulation yet.',
•          BO,SK,SK)
  CALL DISPLAY(14,10,'You need to run an Active simulation first. ,
•          BO,SK,SK)
  CALL DISPLAY(14,54,'<Hit Return>',BO,SK,SK)
  CALL ACCEPT(14,66,HOLD,SK,SK,SK)
  STOP
END IF

```

```

ROW = 0
ROW1 = 0

DO 20 I=1,NUMFILES
  IF (NUMFILES .GT. 10) THEN
    ROW = ROW + 1
  ELSE
    ROW = ROW + 2
  END IF

  IF (I .GT. 20) THEN
    ROW1 = ROW1 + 1
    CALL DISPLAY_INTEGER(ROW1,46,I,3,B0,SK,SK)
    CALL DISPLAY(ROW1,50,'-',SK,SK,SK)
    CALL DISPLAY(ROW1,52,DIS(I),SK,SK,SK)
  ELSE
    CALL DISPLAY_INTEGER(ROW,10,I,3,B0,SK,SK)
    CALL DISPLAY(ROW,14,'-',SK,SK,SK)
    CALL DISPLAY(ROW,16,DIS(I),SK,SK,SK)
  END IF

20  CONTINUE

CALL DISPLAY(23,10,'Select Active loss file to use:',SK,SK,SK)

30  CONTINUE
CALL ACCEPT_INTEGER(23,42,ANS,2,B0,SC,SK)

IF (ANS .LT. 1 .OR. ANS .GT. NUMFILES) THEN
  CALL DISPLAY(23,45,'Invalid input <Hit Return>',B0,NE,BE)
  CALL ACCEPT(23,72,HOLD,SK,SK,SK)
  CALL DISPLAY(23,45,'',SK,SK,SK)
  GOTO 30
END IF

FILEN = STORE(ANS)
ACTFILE = DIS(ANS)

999 CONTINUE

RETURN
END

```

---

SUBROUTINE ACCEPT OUTYEAR RECRUITS

---

This subroutine is set up to allow the user to input via the terminal the new recruits for the SAM program and the OTHER program for each of the outyears. Also, the Active simulation losses will be read into the program.

ROW, I, J, K, START, FINISH      These variables are used as indexes into arrays and as pointers to positions on the terminal.

SAMS(OUTY)      This array is used to store the numbers input by the user for each outyear. The numbers are for new SAM recruits.

OTHER(OUTY)      This array contains the user inputs for new recruits into the OTHER program.

FLAG      This is used for flagging whether the user is inputting for SAMS or OTHER's.

OUTYEAR(OUTY+1,LOS,RATE,PROG)      This array will store the most recent year of history in the first outyear dimension. The other outyear dimensions will be filled by simulating the history forward into the future. This subroutine will be used for filling in the LOS 1 for each outyear.

LOSSES(OUTYEAR,RATE,PROG-2)      This array will be read into the program in this subroutine and it contains the Active simulation losses for each outyear.

ALLOC(RATE,PROG-2)      This array will be read into this subroutine and it contains the percents to show how active losses affiliate into the reserves.

ALLOC\_SAMS(RATE)      This array is read in this subroutine and it contains the percents to show how user inputted SAM's allocate over ratings.

ALLOC\_OTHER(RATE)      This array is read in this subroutine and it contains the percents to show how user inputted OTHER are allocated over ratings.

---

SUBROUTINE ACCEPT\_OUTYEAR\_RECRUITS(OUTYEAR,SAMS,OTHER,LOSSES)

IMPLICIT NONE

```

      INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
      INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS_RES.FOR'

      INTEGER ROW, I, J, K, START, FINISH, SAMS(OUTY), OTHER(OUTY),
      FLAG, ANS

      REAL ALLOC(20,RATE,PROG-2), ALLOC_SAMS(20,RATE),
      ALLOC_OTHER(20,RATE), LOSSES(OUTY,RATE,PROG-2),
      OUTYEAR(OUTY+1,LOS,RATE,PROG)

      CHARACTER HOLD
      CHARACTER*2 CHAR_NUM
      CHARACTER*8 RATE_LABEL
      CHARACTER*50 FILEN

      OPEN (UNIT=51,STATUS='OLD',READONLY,FORM='UNFORMATTED',
      FILE='CNA2:[CORLISSG.FORCE.DAT_RES]ALLOC_RES.DAT')
      READ(51) ALLOC, ALLOC_SAMS, ALLOC_OTHER
      CLOSE (UNIT=51)

1      CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      CALL DISPLAY(10,15,'1',B0,SK,SK)
      CALL DISPLAY(10,16,' - Use the default SAM allocation',
      SK,SK,SK)
      CALL DISPLAY(12,15,'2',B0,SK,SK)
      CALL DISPLAY(12,16,' - Select a user created SAM allocation',
      SK,SK,SK)
      CALL DISPLAY(15,20,'Enter your selection',SK,SK,SK)

3      CONTINUE

      FILEN = '

      CALL ACCEPT_INTEGER(15,42,ANS,1,B0,SC,SK)

      IF (ANS .NE. 1 .AND. ANS .NE. 2) THEN
          CALL DISPLAY(15,45,'Invalid input <Hit Return>',B0,NE,BE)
          CALL ACCEPT(15,71,HOLD,SK,SK,SK)
          CALL DISPLAY(15,45,'',SK,SK,SK)
          GOTO 3
      ELSE IF (ANS .EQ. 2) THEN
          CALL FIND_SAM_ALLOCATION_FILES(FILEN)
      END IF

      IF (FILEN .EQ. 'FLAG') THEN
          GOTO 1
      ELSE IF (ANS .EQ. 2) THEN
          OPEN (UNIT=51,STATUS='OLD',READONLY,FILE=FILEN,
          FORM='UNFORMATTED')
          READ(51) ALLOC_SAMS
          CLOSE (UNIT=51)
      END IF

      FLAG = 0

```



```

5      CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      IF (FLAG .EQ. 0) THEN
          CALL DISPLAY (8,2,'INPUT FOR SAMS',BO,NE,BL)
      ELSE
          CALL DISPLAY (8,2,'INPUT FOR OTHERS',BO,NE,BL)
      END IF

      ROW = 2
      I = 1

      CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
      CALL DISPLAY(ROW,21,' - Current FY',SK,SK,SK)
      CALL DISPLAY(ROW,34,' Total Recruits:',SK,SK,SK)
      CALL DISPLAY(ROW,51,' ',BO,SC,SK)

      DO 10 I=2,OUTY
          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF
          CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
          CALL DISPLAY(ROW,21,' - Outyear',SK,SK,SK)
          CALL DISPLAY_INTEGER(ROW,32,I-1,2,SK,SK,SK)
          CALL DISPLAY(ROW,34,' Total Recruits:',SK,SK,SK)
          CALL DISPLAY(ROW,51,' ',BO,SC,SK)
10     CONTINUE

      IF (OUTY .GT. 10) THEN
          ROW = 1
      ELSE
          ROW = 0
      END IF

      START = 1
      FINISH = OUTY

20     CONTINUE

      DO 30 I=START,FINISH
          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF
          CALL DISPLAY(ROW,51,' ',BO,SC,SK)
          IF (FLAG .EQ. 0) THEN
              SAMS(I) = 0
              CALL ACCEPT_INTEGER(ROW,51,SAMS(I),6,BO,SC,SK)
          ELSE
              OTHER(I) = 0
              CALL ACCEPT_INTEGER(ROW,51,OTHER(I),6,BO,SC,SK)
          END IF
30     CONTINUE

```

```

5      CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      IF (FLAG .EQ. 0) THEN
          CALL DISPLAY (8,2,'INPUT FOR SAMS',BO,NE,BL)
      ELSE
          CALL DISPLAY (8,2,'INPUT FOR OTHERS',BO,NE,BL)
      END IF

      ROW = 2
      I = 1

      CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
      CALL DISPLAY(ROW,21,' - Current FY',SK,SK,SK)
      CALL DISPLAY(ROW,34,' Total Recruits:',SK,SK,SK)
      CALL DISPLAY(ROW,51,' ',BO,SC,SK)

      DO 10 I=2,OUTY
          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF
          CALL DISPLAY_INTEGER(ROW,19,I,2,BO,SK,SK)
          CALL DISPLAY(ROW,21,' - Outyear',SK,SK,SK)
          CALL DISPLAY_INTEGER(ROW,32,I-1,2,SK,SK,SK)
          CALL DISPLAY(ROW,34,' Total Recruits:',SK,SK,SK)
          CALL DISPLAY(ROW,51,' ',BO,SC,SK)
10     CONTINUE

      IF (OUTY .GT. 10) THEN
          ROW = 1
      ELSE
          ROW = 0
      END IF

      START = 1
      FINISH = OUTY

20     CONTINUE

      DO 30 I=START,FINISH
          IF (OUTY .GT. 10) THEN
              ROW = ROW + 1
          ELSE
              ROW = ROW + 2
          END IF
          CALL DISPLAY(ROW,51,' ',BO,SC,SK)
          IF (FLAG .EQ. 0) THEN
              SAMS(I) = 0
              CALL ACCEPT_INTEGER(ROW,51,SAMS(I),6,BO,SC,SK)
          ELSE
              OTHER(I) = 0
              CALL ACCEPT_INTEGER(ROW,51,OTHER(I),6,BO,SC,SK)
          END IF
30     CONTINUE

```

```

      CALL DISPLAY(22,14,'ENTER NUMBER OF OUTYEAR TO CHANGE OR <RETURN>'
                  ,SK,SK,SK)

      CALL DISPLAY(22,60,'      ',BO,SC,SK)
      CALL ACCEPT_INTEGER(22,60,START,2,BO,SC,SK)

      IF ((START .GT. 0) .AND. (START .LE. OUTY)) THEN
        IF (OUTY .GT. 10) THEN
          ROW = START
        ELSE
          ROW = (START - 1) * 2
        END IF

        FINISH = START

        GOTO 20
      END IF

      IF (FLAG .EQ. 0) THEN
        FLAG = 1
        GOTO 5
      END IF

      DO 40 J=1,RATE
        DO 40 I=2,OUTY+1

          OUTYEAR(I,1,J,1) = LOSSES(I-1,J,1) * ALLOC(I,J,1)
          OUTYEAR(I,1,J,2) = LOSSES(I-1,J,2) * ALLOC(I,J,2)

          OUTYEAR(I,1,J,5) = (SAMS(I-1)* 0.9) * ALLOC_SAMS(I,J)
          OUTYEAR(I,1,J,6) = (OTHER(I-1)* 0.9) * ALLOC_OTHER(I,J)

40      CONTINUE

      CALL CHANGE_SCREEN(CL,SK,SK)

      RETURN
      END

```

```

C-----
C
C   SUBROUTINE   FIND   SAM   ALLOCATION   FILES
C
C-----
C   This subroutine is used to allow the user to select any previously
C   created and saved SAM allocation matrix. The subroutine searches the
C   users directory for allocation files and displays to the terminal the
C   choices. If no choices are to be had, 'FLAG' is put in FILEN.
C
C   FILEN                                This is passed back to the main program
C                                       and it contains the filename of the
C                                       users choice or 'FLAG'.
C
C   NEWFILE                              This is used in the library call and
C                                       it contains the next file found in the
C                                       directory. When no more files it
C                                       passes back the default filename.
C
C   DEFAULT, RELEATED, FILENAME          These are used by the library and are
C                                       used to locate a certian set of files.
C
C   STORE(100)                          This is used to store the names of all
C                                       the user created allocations found
C                                       in the directory.
C
C   DIS(100)                             This stores the names of the files to
C                                       be displayed to the user on the screen.
C
C   CHECK                                Is used to check and see when there are
C                                       no more files.
C
C   CONTEXT                              Is used by the library subroutine as an
C                                       address pointer and must be set to 0
C                                       at start.
C
C   ROW, ROW1                             Used to calc. which row of the screen
C                                       to display on.
C
C   BEGIN, END                            Used to store the begin and end of
C                                       strings that are being searched for in
C                                       other strings.
C
C   NUMFILES                             Is a counter used to keep count of the
C                                       of allocations found in the directory.
C
C   ANS                                  The answer to which allocation the user
C                                       selected to use.
C
C   I, DIFF                              Used as integers for loops and
C                                       calculations.
C-----

```

```

SUBROUTINE FIND_SAM_ALLOCATION_FILES(FILEN)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG FORCE]SCREEN_PARAMETERS.FOR'

INTEGER CONTEXT, ROW, ROW1, BEGIN, END, NUMFILES, ANS, I, DIFF

CHARACTER HOLD

CHARACTER*50 FILENAME, NEWFILE, DEFAULT, RELATED, OLDFILE,
*      STORE(100), DIS(100), CHECK, FILEN

CALL CHANGE_SCREEN(CL,SK,SK)

FILENAME = 'SAM_*.DAT'

DEFAULT = FILENAME
RELATED = FILENAME

CONTEXT = 0
NUMFILES = 1

CHECK = '

10  CONTINUE

CALL LIB$FIND_FILE(FILENAME,NEWFILE,CONTEXT,DEFAULT,RELATED)

BEGIN = INDEX(NEWFILE,']SAM_')
END = INDEX(NEWFILE,'.DAT')
BEGIN = BEGIN + 1
END = END + 3
DIFF = (END - BEGIN) + 1

CHECK(1:DIFF) = NEWFILE(BEGIN:END)
DIS(NUMFILES) = CHECK(5:DIFF-4)

IF (CHECK .NE. FILENAME) THEN
    IF (NEWFILE .NE. STORE(NUMFILES)) THEN
        STORE(NUMFILES) = NEWFILE
        NUMFILES = NUMFILES + 1
    END IF
    CHECK = '
    GOTO 10
END IF

NUMFILES = NUMFILES - 1

IF (NUMFILES .EQ. 0) THEN
    CALL DISPLAY(10,10,'No user created allocations yet <Hit Return>'
    ,80,BE,SK)
    CALL ACCEPT(10,55,HOLD,SK,SK,SK)
    FILEN = 'FLAG'
    GOTO 999
END IF

ROW = 0
ROW1 = 0

DO 20 I=1,NUMFILES

```

```

        IF (NUMFILES .GT. 10) THEN
            ROW = ROW + 1
        ELSE
            ROW = ROW + 2
        END IF

        IF (I .GT. 20) THEN
            ROW1 = ROW1 + 1
            CALL DISPLAY_INTEGER(ROW1,46,I,3,B0,SK,SK)
            CALL DISPLAY(ROW1,50,'-',SK,SK,SK)
            CALL DISPLAY(ROW1,52,DIS(I),SK,SK,SK)
        ELSE
            CALL DISPLAY_INTEGER(ROW,10,I,3,B0,SK,SK)
            CALL DISPLAY(ROW,14,'-',SK,SK,SK)
            CALL DISPLAY(ROW,16,DIS(I),SK,SK,SK)
        END IF

20      CONTINUE

        CALL DISPLAY(23,10,'Enter your selection:',SK,SK,SK)

30      CONTINUE
        CALL ACCEPT_INTEGER(23,32,ANS,2,B0,SC,SK)

        IF (ANS .LT. 1 .OR. ANS .GT. NUMFILES) THEN
            GOTO 30
        END IF

        FILEN = STORE(ANS)

999     CONTINUE

        RETURN
        END

```

```

C-----
C
C           S U B R O U T I N E   C H A N G E   M I X E S
C-----
C
C      This subroutine controls which year the user wants to change the
C      SAM allocation.
C
C      I, J, K, ROW           These variables are used as indexes
C                             into arrays and as pointers to
C                             positions on the screen for displaying.
C
C      YEAR                   Holds the years the user has selected
C                             to work with.  Index into arrays.
C
C      DIS_YEAR               Stores the years the user is working
C                             with.  For use in displaying to screen.
C
C      DONE(OUTY)             This matrix stores flags to indicate
C                             which of the outyears have been worked
C                             on.
C
C      OUTYEAR(OUTY+1,LOS,RATE,PROG) This matrix is storing the history and
C                             LOS 1 for each of the outyears.  This
C                             is the matrix that is changed by the
C                             user.
C-----

```

```

SUBROUTINE CHANGE_MIXES(OUTYEAR)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS_RES.FOR'

```

```

INTEGER I, J, K, YEAR, DIS_YEAR, ROW, DONE(OUTY)

```

```

REAL OUTYEAR(OUTY+1,LOS,RATE,PROG)

```

```

CHARACTER HOLD, ANS

```

```

C-----
C  First check to see if any changes are required.
  CALL CHANGE_SCREEN(CL,SK,SK)
  CALL DISPLAY(13,10,'Would you like to change Sam mixes (' ,SK,SK,SK)
  CALL DISPLAY(13,46,'Y' ,BO,SK,SK)
  CALL DISPLAY(13,47,'es or' ,SK,SK,SK)
  CALL DISPLAY(13,53,'N' ,BO,SK,SK)
  CALL DISPLAY(13,54,'o)?' ,SK,SK,SK)
5  CONTINUE
  CALL ACCEPT(13,58,ANS,BO,SC,SK)
  IF (ANS .NE. 'N' .AND. ANS .NE. 'Y' .AND. ANS .NE. 'n' .AND.
      ANS .NE. 'y') THEN
      GOTO 5
  END IF

```

C If no changes then exit this subroutine.

```
IF (ANS .EQ. 'N' .OR. ANS .EQ. 'n') THEN
    GOTO 999
END IF
```

C-----

10 CONTINUE

CALL CHANGE\_SCREEN(CL,SK,SK)

ROW = 2

I = 1

DIS\_YEAR = CFY

CALL DISPLAY\_INTEGER(ROW,24,I,2,B0,SK,SK)

CALL DISPLAY(ROW,27,'- Current FY',SK,SK,SK)

CALL DISPLAY\_INTEGER(ROW,40,DIS\_YEAR,4,SK,SK,SK)

IF (DONE(1) .EQ. 1) THEN

CALL DISPLAY(ROW,21,'X',B0,NE,SK)

END IF

DO 20 I=2,OUTY

DIS\_YEAR = DIS\_YEAR + 1

IF (OUTY .LE. 10) THEN

ROW = ROW + 2

ELSE

ROW = ROW + 1

END IF

CALL DISPLAY\_INTEGER(ROW,24,I,2,B0,SK,SK)

CALL DISPLAY(ROW,27,'- Outyear',SK,SK,SK)

CALL DISPLAY\_INTEGER(ROW,37,I-1,2,SK,SK,SK)

CALL DISPLAY\_INTEGER(ROW,40,DIS\_YEAR,4,SK,SK,SK)

IF (DONE(1) .EQ. 1) THEN

CALL DISPLAY(ROW,21,'X',B0,NE,SK)

END IF

20 CONTINUE

CALL DISPLAY(22,20,'Enter Your Selection or (',SK,SK,SK)

CALL DISPLAY(22,45,'99',B0,SK,SK)

CALL DISPLAY(22,48,'to end):',SK,SK,SK)

30 CONTINUE

C-----

C Accept the year and check to see if good

C selection or finished.

CALL ACCEPT\_INTEGER(22,57,YEAR,2,B0,SC,SK)

IF (YEAR .EQ. 99) THEN

GOTO 999

ELSE IF (YEAR .LT. 1 .OR. YEAR .GT. OUTY) THEN

CALL DISPLAY(22,60,'Bad input Hit Return',B0,NE,BE)

CALL ACCEPT(22,80,HOLD,SK,SK,SK)

CALL DISPLAY(22,60,'',SK,SK,SK)



```
          GOTO 30
        END IF

        CALL CHANGE_SAM_MIX(YEAR,OUTYEAR,DONE)

        GOTO 10

999      CONTINUE

        CALL CHECK_TO_SEE_IF_SAVE_NEW_ALLOC(OUTYEAR)

        RETURN
      END
```

---

S U B R O U T I N E   C H A N G E   S A M   M I X

---

This subroutine allows the user to change the SAM allocation by Rating for the year selected. In this subroutine the user will choose the Ratings to work with.

YEAR, YR	Used to store the year the user has selected to work with. Is passed to this subroutine.
START, FINISH, ROW	These are used to compute where on the screen the cursor should be and to store which rating to work with.
I, J, K	Used as indexes into arrays.
CNT	Used to count the number of changes.
DIFF, NUM	Used in computing and displaying
DONE(OUTY)	If a change is made then flag the change in the appropriate year
ORIG(RATE+1)	Stores the way the SAMs are allocated upon entering this subroutine.
D(RATE+1)	Will store the differences in changes made. Has to = 0 before exiting this subroutine.
PROGRAMB(RATE+1)	This is the matrix the user changes around. After finished this array is used to update the OUTYEAR matrix.
NUM_RATING	Has the number of rating selected to work with.
RATING_INDEX(RATE)	Stores the ratings the user selected to change.
OUTYEAR(OUTY+1, LOS, RATE, PROG)	Is updated at end of subroutine with the changes made.

---

SUBROUTINE CHANGE\_SAM\_MIX(YR,OUTYEAR,DONE)

IMPLICIT NONE

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN\_PARAMETERS.FOR'  
 INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS\_RES.FOR'

INTEGER YEAR, YR, START, FINISH, I, J, K, ORIG(RATE+1),  
 CNT, DIFF, ROW, PROGRAMB(RATE+1), D(RATE+1).

```

      NUM, DONE(OUTY), NUM_RATING, RATING_INDEX(RATE)

      REAL OUTYEAR(OUTY+1,LOS,RATE,PROG)

      CHARACTER HOLD

      CHARACTER*8 RATE_LABEL

      CNT = 0
      YEAR = YR + 1

C-----
C  Calling this subroutine allows the user to select
C  the ratings to work with.
      CALL CHOOSE_RATING(RATING_INDEX, NUM_RATING)
C-----

C-----
C  Only 16 ratings can be worked with at one time.
      IF (NUM_RATING .GT. 16) THEN
          NUM_RATING = 16
      END IF
C-----

      START = 1
      FINISH = NUM_RATING
      ROW = 2

C-----
C  The first thing to do is store what the LOS 1
C  looks like by RATING for the selected outyear.
C  The 10 loop loads in LOS 1 and computes a total accession.
      ORIG(NUM_RATING+1) = 0
      PROGRAMB(NUM_RATING+1) = 0
      D(NUM_RATING+1) = 0

      DO 10 I=1,NUM_RATING
          ORIG(I) = 0
          PROGRAMB(I) = 0
          D(I) = 0
          ORIG(I) = OUTYEAR(YEAR,1,RATING_INDEX(I),5) + .5
          PROGRAMB(I) = OUTYEAR(YEAR,1,RATING_INDEX(I),5) + .5
          ORIG(NUM_RATING+1) = ORIG(NUM_RATING+1) + ORIG(I)
10      PROGRAMB(NUM_RATING+1) = PROGRAMB(NUM_RATING+1) + PROGRAMB(I)
C-----

      CALL CHANGE_SCREEN(CL,SK,SK)

      IF (NUM_RATING .GT. 8) THEN
          CALL DISPLAY (1,29,'Orig',SC,SK,SK)
          CALL DISPLAY (1,40,'Change',SC,SK,SK)
          CALL DISPLAY (1,52,'Diff',SC,SK,SK)
      ELSE
          CALL DISPLAY (2,29,'Orig',SC,SK,SK)
          CALL DISPLAY (2,40,'Change',SC,SK,SK)
          CALL DISPLAY (2,52,'Diff',SC,SK,SK)
      END IF

```

```

DO 20 I=START,FINISH
    IF (NUM_RATING .GT. 8) THEN
        ROW = ROW + 1
    ELSE
        ROW = ROW + 2
    END IF
    CALL DISPLAY_INTEGER(ROW,10,1,2,BO,SK,SK)
    CALL DISPLAY(ROW,13,RATE_LABEL(RATING_INDEX(I)),SK,SK,SK)
    CALL DISPLAY_INTEGER(ROW,27,ORIG(I),6,BO,SK,SK)
    CALL DISPLAY_INTEGER(ROW,40,PROGRAMB(I),6,BO,SC,SK)
    CALL DISPLAY_INTEGER(ROW,50,D(I),6,BO,SK,SK)
20  CONTINUE

    ROW = ROW + 2
    CALL DISPLAY(ROW,13,RATE_LABEL(RATE+1),SK,SK,SK)
    CALL DISPLAY_INTEGER(ROW,27,ORIG(NUM_RATING+1),6,BO,SK,SK)
    CALL DISPLAY_INTEGER(ROW,40,PROGRAMB(NUM_RATING+1),6,BO,SC,SK)
    CALL DISPLAY_INTEGER(ROW,50,D(NUM_RATING+1),6,BO,SK,SK)

    CALL DISPLAY(22,10,'Enter Rating to change or <Return>:',SK,SK,SK)
30  CONTINUE

    CALL ACCEPT_INTEGER(22,47,START,2,BO,SC,SK)

C-----
C  After a RATING is selected to change the user inputs
C  the new number and the difference is computed and displayed.
C  If a change is made an equal change must be made in some other
C  RATING to offset it.
    IF (START .GT. 0 .AND. START .LT. NUM_RATING+1) THEN
        CNT = CNT + 1
        IF (NUM_RATING .GT. 8) THEN
            ROW = 2 + START
        ELSE
            ROW = 2 + (START * 2)
        END IF
        NUM = PROGRAMB(START)
        CALL ACCEPT_INTEGER(ROW,40,PROGRAMB(START),6,BO,SC,SK)
        D(START) = D(START) + (PROGRAMB(START) - NUM)
        D(NUM_RATING+1) = D(NUM_RATING+1) + (PROGRAMB(START) - NUM)
        PROGRAMB(NUM_RATING+1) = PROGRAMB(NUM_RATING+1) +
            (PROGRAMB(START) - NUM)
        CALL DISPLAY_INTEGER(ROW,50,D(START),6,BO,SK,NE)
        IF (NUM_RATING .GT. 8) THEN
            ROW = 4 + NUM_RATING
        ELSE
            ROW = 4 + (NUM_RATING * 2)
        END IF
        CALL DISPLAY_INTEGER(ROW,40,PROGRAMB(NUM_RATING+1),6,BO,SK,NE)
        CALL DISPLAY_INTEGER(ROW,50,D(NUM_RATING+1),6,BO,SK,NE)
        GOTO 30
    END IF

C-----
C  If no changes were made then exit subroutine.
    IF (CNT .EQ. 0) THEN

```

```

        GOTO 999
    END IF

    DONE(YEAR-1) = 1

    DIFF = PROGRAMB(NUM_RATING+1) - ORIG(NUM_RATING+1)

C-----
C  This IF statement is checking to make sure that before
C  exiting the total is the same as when entering the subroutine.
C  If not an error message is sent to the user and more changes
C  must be made.
        IF (DIFF .GT. 0 .OR. DIFF .LT. 0) THEN
            CALL DISPLAY(23,10,'The Total has to =',B0,NE,BE)
            CALL DISPLAY_INTEGER(23,29,ORIG(NUM_RATING+1),6,B0,SC,SK)
            CALL DISPLAY(23,36,'not',B0,NE,SK)
            CALL DISPLAY_INTEGER(23,40,PROGRAMB(NUM_RATING+1),6,B0,SC,SK)
            CALL DISPLAY(23,49,'<Hit Return>',B0,NE,SK)
            CALL ACCEPT(23,62,HOLD,SK,SK,SK)
            CALL DISPLAY(23,10,'
*           ,SK,SK,SK)
            CALL DISPLAY(23,45,'
*           ,SK,SK,SK)
            GOTO 30
        END IF

999    CONTINUE

    DO 40 I=1,NUM_RATING
40      OUTYEAR(YEAR,1,RATING_INDEX(I),5) = PROGRAMB(I)

    CALL CHANGE_SCREEN(CL,SK,SK)

    RETURN
    END

```

```

C-----
C
C           S U B R O U T I N E   C H E C K   T O   S E E   I F   N E W   A L L O C
C
C-----
C
C      This subroutine asks the user if the SAM allocation matrix being used
C      in this run should be saved on disk or not.
C
C      OUTYEAR(OUTY+1,LOS,RATE,PROG)           This matrix has the LOS 1
C                                              filled in for each of the
C                                              outyears with changes made to
C                                              SAMS. The SAMS allocations for
C                                              each outyear will be extracted
C                                              to create a new allocation.
C
C      NEW_SAM(20,RATE)                       This matrix will store the new
C                                              SAM allocation to be saved.
C
C      TOT(OUTY)                               This stores the sum of all the
C                                              ratings for each outyear. It
C                                              is the denominator.
C-----

```

```

SUBROUTINE CHECK_TO_SEE_IF_SAVE_NEW_ALLOC(OUTYEAR)

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS.FOR'

REAL OUTYEAR(OUTY+1,LOS,RATE,PROG), NEW_SAM(20,RATE), TOT(OUTY)

CHARACTER ANS

CHARACTER*50 FILEN
CHARACTER*25 NAME

CALL CHANGE_SCREEN(CL,SK,SK)

CALL DISPLAY(10,10,'Would you like to save this allocation (',
              SK,SK,SK)
CALL DISPLAY(10,50,'Y',B0,SK,SK)
CALL DISPLAY(10,51,'es or',SK,SK,SK)
CALL DISPLAY(10,57,'N',B0,SK,SK)
CALL DISPLAY(10,58,'o)?',SK,SK,SK)

5  CONTINUE
CALL ACCEPT(10,62,ANS,B0,SC,SK)
IF (ANS .NE. 'N' .AND. ANS .NE. 'Y' .AND. ANS .NE. 'n' .AND.
    ANS .NE. 'y') THEN
    GOTO 5
END IF

IF (ANS .EQ. 'N' .OR. ANS .EQ. 'n') THEN
    GOTO 999
END IF

CALL DISPLAY(14,5,'Enter a name to save this allocation under',

```

```

      SK,SK,SK)

10  CONTINUE
    NAME = '
    CALL ACCEPT(14,48,NAME,BO,NE,SK)

    IF (NAME .EQ. '          ') THEN
      GOTO 10
    END IF

    FILEN = 'SAM_'//NAME

    OPEN(UNIT=60,STATUS='NEW',FILE=FILEN,FORM='UNFORMATTED',ERR=10)

C   Compute the total over ratings for each outyear.  Creating the denominator
    DO 20 J=1,RATE
      DO 20 I=1,OUTY
20      TOT(I) = TOT(I) + OUTYEAR(I+1,1,J,5)

C   Computing the new SAM allocation matrix.
    DO 30 J=1,RATE
      DO 30 I=1,OUTY
        IF (TOT(I) .GT. 0.0) THEN
          NEW_SAM(I,J) = OUTYEAR(I+1,1,J,5) / TOT(I)
        ELSE
          NEW_SAM(I,J) = 0.0
        END IF
30      CONTINUE

C   Making each year after OUTY equal to OUTY, up to 20 outyears.
    DO 40 J=1,RATE
      DO 40 I=OUTY+1,20
40      NEW_SAM(I,J) = NEW_SAM(OUTY,J)

    WRITE(60) NEW_SAM
    CLOSE(UNIT=60)

999  CONTINUE

    RETURN
    END

```

```

C-----
C
C           S U B R O U T I N E   T R A N S I S T I O N
C
C-----
C
C      This is the main subroutine of the program.  This subroutine is set
C      up to take the history from the most recent year, read it into the
C      program, then transition it using the percents read into the program,
C      to create the outyear inventories.
C
C      I, J, K, L           These variables are used as indexes
C                           into arrays.
C
C      TRANS(LOS,RATE,PROG) This array is read into this subroutine
C                           and it contains the continuation rates
C                           which are used to transition the
C                           history.
C
C      HIST(LOS,RATE,PROG)  This array is read into this subroutine
C                           and contains the history for the
C                           most recent year.  This data will be
C                           loaded into the first outyear dimen. of
C                           the OUTYEAR array.
C
C      OUTYEAR(OUTY+1,LOS,RATE,PROG) OUTYEAR coming into this subroutine
C                                    contains only the LOS 1 for each of the
C                                    outyears.  In this subroutine the
C                                    History will be loaded into the outyear
C                                    1 dimension and the the transition rates
C                                    will be applied to fill up this array.
C                                    This is where the simulated results are
C                                    stored.
C-----

```

```

SUBROUTINE TRANSISTION(OUTYEAR)

IMPLICIT NONE

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'
INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS_RES.FOR'

INTEGER I, J, K, L

REAL TRANS(LOS,RATE,PROG), OUTYEAR(OUTY+1,LOS,RATE,PROG),
•   HIST(LOS,RATE,PROG)

CHARACTER*2 CHAR_NUM
CHARACTER*25 FILEN

CALL CHANGE_SCREEN(CL,SK,SK)

CALL DISPLAY(10,15,'Please wait while simulation takes place',
•   BO,NE,SK)

OPEN(UNIT=59,STATUS='OLD',READONLY,FORM='UNFORMATTED',

```



```

      • FILE='CNA2:[CORLISSG.FORCE.DAT_RES]TRANS_RES.DAT')
      READ(59) TRANS
      CLOSE(UNIT=59)

      OPEN(UNIT=59,STATUS='OLD',READONLY,FORM='UNFORMATTED',
      • FILE='CNA2:[CORLISSG.FORCE.DAT_RES]HIST_RES.DAT')
      READ(59) HIST
      CLOSE(UNIT=59)

      DO 10 J=1,LOS
          DO 10 K=1,RATE
              DO 10 L=1,PROG
10          OUTYEAR(1,J,K,L) = HIST(J,K,L)

      DO 30 I=1,OUTY
          DO 30 L=1,PROG
              DO 30 K=1,RATE
                  DO 20 J=1,LOS-1

                  OUTYEAR(I+1,J+1,K,L) = OUTYEAR(I,J,K,L) * TRANS(J,K,L)

20          CONTINUE

          OUTYEAR(I+1,LOS,K,L) = OUTYEAR(I+1,LOS,K,L) +
      • OUTYEAR(I,LOS,K,L) * TRANS(LOS,K,L)

30      CONTINUE

      OPEN(UNIT=59,STATUS='NEW',FORM='UNFORMATTED',
      • FILE='OUTYEAR_RES.DAT')
      WRITE(59) OUTYEAR
      CLOSE(UNIT=59)

      CALL CHANGE_SCREEN(CL,SK,SK)

      RETURN
      END

```

---

S U B R O U T I N E   A D D   P A Y G R A D E

---

This subroutine is used to restructure the OUTYEAR array so it can be compared to requirements. The OUTYEAR array will be collapsed over program and LOS, and paygrade will be added.

I, J, K, L, M                      These variables are used as indexes into arrays.

OUTYEAR(OUTY+1,LOS,RATE,PROG)    This array is passed into this subroutine and it contains the simulated results.

PAYGRADE(LOS,RATE,PROG,PAYGRADE) This array is read into this subroutine and it contains percents which show how paygrade breaks out of the simulated inventory.

SUM(OUTY,RATE,PAYGRADE)           This array is created by collapsing out Program and LOS and adding paygrade to the outyear inventory. Used for comparing to requirements.

---

SUBROUTINE ADD\_PAYGRADE(OUTYEAR,SUM)

IMPLICIT NONE

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN\_PARAMETERS.FOR'

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS\_RES.FOR'

INTEGER I, J, K, L, M

REAL OUTYEAR(OUTY+1,LOS,RATE,PROG), PAYGRADE(LOS,RATE,PROG,PAYG),  
 SUM(OUTY,RATE,PAYG)

CHARACTER\*2 CHAR\_NUM

CHARACTER\*25 FILEN

CALL CHANGE\_SCREEN(CL,SK,SK)

CALL DISPLAY(10,15,'Please wait while paygrade added',BO,SK,SK)

OPEN (UNIT=56,STATUS='OLD',READONLY,FORM='UNFORMATTED',  
 FILE='CNA2:[CORLISSG.FORCE.DAT\_RES]PAYGRADE\_RES.DAT')

READ(56) PAYGRADE

CLOSE(UNIT=56)

DO 10 M=1,PAYG

    DO 10 L=1,PROG

        DO 10 K=1,RATE

            DO 10 J=1,LOS

                DO 10 I=2,OUTY+1

```
10      SUM(I-1,K,M) = SUM(I-1,K,M) +  
      (OUTYEAR(I,J,K,L) * PAYGRADE(J,K,L,M))  
  
      CALL CHANGE_SCREEN(CL,SK,SK)  
  
      RETURN  
      END
```

---

S U B R O U T I N E   W R I T E I T

---

This subroutine is used to write out to the screen and to a disk file the results of the simulation. The simulated results will be compared to the NAVY requirements.

RATING\_INDEX(RATE)                      This array is used to store which ratings the user has selected to view on the terminal.

NUM\_RATING                                This is a count of how many ratings the user selected to view.

I, J, K, L                                These variables are used as indexes into arrays.

OUT                                        This stores the year that the user selected for viewing on the terminal.

CNT                                        This is used to keep a count of where a page break is needed in the output.

SAMS(OUTY)                                This array is used to store the numbers input by the user for each outyear. The numbers are for new SAM recruits.

OTHER(OUTY)                                This array contains The user inputs for new recruits into the OTHER program.

LOSSES(OUTYEAR,RATE,PROG-2)              This array was read into the program and it contains the Active Simulation losses for each outyear.

TOT(PROG-2)                                This array is used to compute the totals for the LOSSES array by program.

REQ(20,RATE,PAYG)                        This array is read into the program in this subroutine and it contains the NAVY requirements.

SUM(OUTY,RATE,PAYG)                        This array is passed into this subroutine and it contains the simulated results for each outyear by Rate and Paygrade.

S(OUTY+1,RATE+1,PAYG+1)                  This array is the same as SUM except that Totals have been added.

SR(OUTY+1,RATE+1,PAYG+1)                  This array is the same as REQ except that Totals have been added.

DIFF(OUTY+1,RATE+1,PAYG+1)                This array is computed by subtracting SR from S in each cell.

```

C
C      RATIO(OUTY+1,RATE+1,PAYG+1)      This array is computed by dividing
C                                         DIFF by SR or every cell where SR > 0 0
C-----

```

```

SUBROUTINE WRITE_RESULTS(SUM,SAMS,OTHER,LOSSES,ACTFILE)

```

```

IMPLICIT NONE

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]SCREEN_PARAMETERS.FOR'

```

```

INCLUDE 'CNA2:[CORLISSG.FORCE]PARAMETERS_RES.FOR'

```

```

INTEGER I, J, K, L, OUT, REQ(20,RATE,PAYG), CNT,

```

```

*      SAMS(OUTY), OTHER(OUTY), RATING_INDEX(RATE), NUM_RATING

```

```

REAL SUM(OUTY,RATE,PAYG), DIFF(OUTY+1,RATE+1,PAYG+1),

```

```

*      RATIO(OUTY+1,RATE+1,PAYG+1), S(OUTY+1,RATE+1,PAYG+1),

```

```

*      SR(OUTY+1,RATE+1,PAYG+1), LOSSES(OUTY,RATE,PROG-2),

```

```

*      TOT(PROG-2)

```

```

CHARACTER HOLD, ANS

```

```

CHARACTER*2 CHAR_NUM, YR_SEL

```

```

CHARACTER*8 RATE_LABEL

```

```

CHARACTER*25 NAME

```

```

CHARACTER*50 FILEN, ACTFILE

```

```

CALL DISPLAY(10,15,'Please wait while results are written to disk'

```

```

*      ,BO,SK,SK)

```

```

C--- Initializing

```

```

DO 5 I=1,OUTY+1

```

```

    DO 5 J=1,RATE+1

```

```

        DO 5 K=1,PAYG+1

```

```

            S(I,J,K) = 0.0

```

```

            SR(I,J,K) = 0.0

```

```

            DIFF(I,J,K) = 0.0

```

```

5          RATIO(I,J,K) = 0.0

```

```

C-----

```

```

OPEN(UNIT=50,FILE='CNA2:[CORLISSG.FORCE.DAT_RES]REQ_RES.DAT',

```

```

*      FORM='UNFORMATTED',STATUS='OLD',READONLY)

```

```

READ(50) REQ

```

```

CLOSE(UNIT=50)

```

```

C-----

```

```

C      The 10 loops will sum up everything needed

```

```

DO 10 K=1,PAYG

```

```

    DO 10 J=1,RATE

```

```

        DO 10 I=1,OUTY

```

```

            S(I,J,K) = S(I,J,K) + SUM(I,J,K)

```

```

            S(I,RATE+1,K) = S(I,RATE+1,K) + SUM(I,J,K)

```

```

            S(I,J,PAYG+1) = S(I,J,PAYG+1) + SUM(I,J,K)

```

```

            S(OUTY+1,J,K) = S(OUTY+1,J,K) + SUM(I,J,K)

```

```

            S(OUTY+1,RATE+1,K) = S(OUTY+1,RATE+1,K) + SUM(I,J,K)

```

```

S(I,RATE+1,PAYG+1) = S(I,RATE+1,PAYG+1) + SUM(I,J,K)
S(OUTY+1,J,PAYG+1) = S(OUTY+1,J,PAYG+1) + SUM(I,J,K)
S(OUTY+1,RATE+1,PAYG+1) = S(OUTY+1,RATE+1,PAYG+1) + SUM(I,J,K)

SR(I,J,K) = SR(I,J,K) + REQ(I,J,K)
SR(I,RATE+1,K) = SR(I,RATE+1,K) + REQ(I,J,K)
SR(I,J,PAYG+1) = SR(I,J,PAYG+1) + REQ(I,J,K)
SR(OUTY+1,J,K) = SR(OUTY+1,J,K) + REQ(I,J,K)
SR(OUTY+1,RATE+1,K) = SR(OUTY+1,RATE+1,K) + REQ(I,J,K)
SR(I,RATE+1,PAYG+1) = SR(I,RATE+1,PAYG+1) + REQ(I,J,K)
SR(OUTY+1,J,PAYG+1) = SR(OUTY+1,J,PAYG+1) + REQ(I,J,K)
SR(OUTY+1,RATE+1,PAYG+1) = SR(OUTY+1,RATE+1,PAYG+1) + REQ(I,J,K)

```

10 CONTINUE

C-----  
C The next section is for creating a disk file that shows the  
C comparison of simulated results to requirements

```
OPEN (UNIT=66,STATUS='NEW',FILE='RESERVE.DAT')
```

```
DO 20 K=1,PAYG+1
```

```
DO 20 J=1,RATE+1
```

```
DO 20 I=1,OUTY+1
```

```
DIFF(I,J,K) = S(I,J,K) - SR(I,J,K)
```

```
IF (SR(I,J,K) .EQ. 0) THEN
```

```
RATIO(I,J,K) = 0.0
```

```
ELSE
```

```
RATIO(I,J,K) = DIFF(I,J,K) / SR(I,J,K)
```

```
END IF
```

20 CONTINUE

```
DO 50 I=1,OUTY
```

```
WRITE(66,120)
```

```
WRITE(66,100)
```

```
WRITE(66,100)
```

```
WRITE(66,110) I + (CFY - 1)
```

```
WRITE(66,100)
```

```
WRITE(66,100)
```

```
WRITE(66,130)
```

```
WRITE(66,100)
```

```
WRITE(66,140) SAMS(I), OTHER(I)
```

```
WRITE(66,100)
```

```
WRITE(66,180) ACTFILE
```

```
WRITE(66,100)
```

```
WRITE(66,150)
```

```
WRITE(66,100)
```

```
WRITE(66,170)
```

```
DO 31 K=1,4
```

31 TOT(K) = 0.0

```
DO 33 J=1,RATE
```

```
WRITE(66,90) RATE_LABEL(J),(LOSSES(I,J,K),K=1,4)
```



```

        WRITE(65,100)
        WRITE(66,90) 'TOT TOT ', S(OUTY+1,RATE+1,PAYG+1),
        *           SR(OUTY+1,RATE+1,PAYG+1), DIFF(OUTY+1,RATE+1,PAYG+1),
        *           RATIO(OUTY+1,RATE+1,PAYG+1)

        CLOSE (UNIT=66)

C-----
C Writing out the sums of the simulated results to disk for use in the
C Compare program.

        CALL DISPLAY(13,5,'Enter a filename for storing the results',
        *           SK,SK,SK)
        CALL DISPLAY(13,46,'(For use later in the compare run)',SK,SK,SK)

200    CONTINUE

        NAME = '
        CALL ACCEPT(15,25,NAME,BO,NE,SK)

        IF (NAME .EQ. '          ') THEN
            GOTO 200
        END IF

        FILEN = 'RES_'//NAME

        OPEN(UNIT=67,STATUS='NEW',FILE=FILEN,FORM='UNFORMATTED')
        WRITE(67) S
        CLOSE(UNIT=67)

90     FORMAT(4X,A8,4F17.3)
100    FORMAT(1X)
110    FORMAT(30X,'Y E A R ',I4)
120    FORMAT('1')
130    FORMAT(22X,'Inputs for the outyear are')
140    FORMAT(20X,'Sams = ',I6,' Others = ',I6)
150    FORMAT(20X,' Active losses by rating are:')
160    FORMAT(12X,'          Simulated',',', ' Requirements',
        *           '          Difference',',', ' Percent Diff')
170    FORMAT(16X,'4YO+5&6YO+PS+TARS',',', ' Active Mariners',
        *           '          Sea College',',', ' Other')
180    FORMAT(15X,'Active loss file being used is: ',A50)

999    CONTINUE

        CALL CHANGE_SCREEN(CL,SK,SK)

        RETURN
        END

```